

NETWORK SYSTEM, INFORMATION PROCESSOR, INFORMATION PROCESSING METHOD AND CONTROL PROGRAM

特許公報番号 JP2002251366

公報発行日 2002-09-06

発明者: MACHIDA HARUO

出願人 CANON KK.

分類:

一国際: G06F13/10; G06F3/00; G06F9/445; G06F13/10; G06F3/00; G06F9/445; (IPC1-7): G06F13/10; G06F9/445

一欧州:

出願番号 JP20010355016 20011120

優先権主張番号: JP20010355016 20011120; JP20000391201 20001222

他の公開

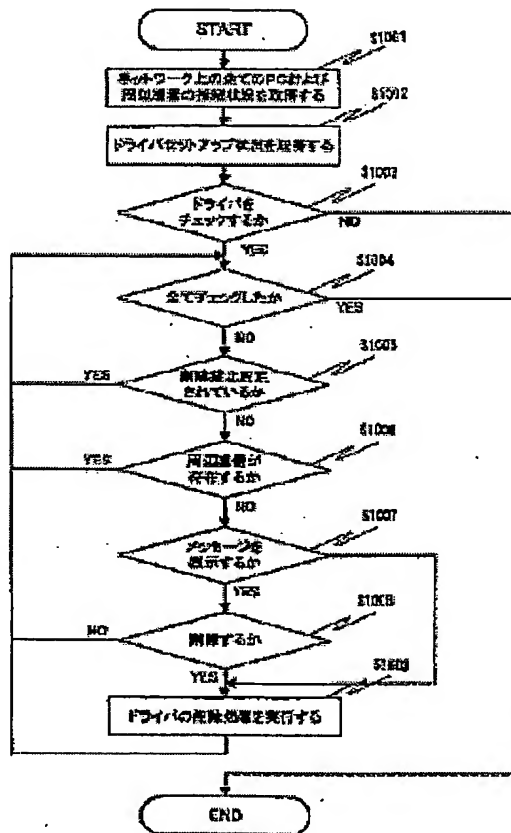
US2002083227 (A1)

ここにデータエラーを報告してください

要約 JP2002251366

PROBLEM TO BE SOLVED: To eliminate a driver which is not required with a simple operation when peripheral equipment shared on a network is eliminated or it cannot be used by allowing shared setting to be released.

SOLUTION: Connection situation information on all PC and peripheral equipment on the network is obtained, and setup situation information of the driver of puerperal equipment installed in a self-machine is obtained. When the check instruction of the driver is given, it is decided whether the elimination inhibition of the driver to be checked is set or not. When it is not inhibited, it is decided whether a device corresponding to the driver exists or not. When the objective device does not exist, a message for urging the user to designate whether elimination is to be executed or not is displayed and the elimination processing of the driver is executed.



esp@cenet データベースから供給されたデータ - Worldwide

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The information processor characterized by having a setup information acquisition means to be the information processor which communicates with a peripheral device, and to acquire the setup situation of the driver in said information processor, and a judgment means to judge whether the peripheral device corresponding to the driver currently set up in said information processor exists.

[Claim 2] Said judgment means is an information processor according to claim 1 characterized by judging whether a peripheral device exists by whether it asks an output destination change port and there is any response.

[Claim 3] It is the information processor according to claim 1 characterized by connecting said information processor to said peripheral-device list through a network.

[Claim 4] The information processor according to claim 1 characterized by having a driver deletion means to delete said driver when judged with the peripheral device corresponding to said driver not existing with said judgment means.

[Claim 5] The information processor according to claim 4 characterized by having a directions means to direct activation of deletion of the driver by said driver deletion means.

[Claim 6] Said driver deletion means is an information processor according to claim 5 characterized by performing deletion of two or more drivers according to one directions by said directions means.

[Claim 7] The information processor according to claim 4 characterized by having a check means to check to a user whether deletion is performed or not before deleting said driver with said driver deletion means.

[Claim 8] The information processor according to claim 4 characterized by having a deletion prohibition setting means to perform a prohibition setup of deletion of a desired driver.

[Claim 9] Said information processor is an information processor according to claim 1 which it communicates with an external information processor and said judgment means judges that the peripheral device corresponding to said driver does not exist, and is characterized by having further a notice means of driver deletion to perform the notice of driver deletion which makes said external information processor delete a driver when said peripheral device is a local device.

[Claim 10] The information processing approach characterized by having the setup information acquisition procedure which is the information processing approach in the information processor which communicates with a peripheral device, and acquires the setup situation of the driver in said information processor, and the judgment procedure of judging whether the peripheral device corresponding to the driver currently set up in said information processor existing.

[Claim 11] The information processing approach according to claim 10 characterized by judging whether a peripheral device exists by whether it asks an output destination change port and there is any response in said judgment procedure.

[Claim 12] It is the information processing approach according to claim 10 characterized by connecting said information processor to said peripheral-device list through a network.

[Claim 13] The information processing approach according to claim 10 characterized by having the driver deletion procedure of deleting said driver when judged with the peripheral device corresponding to said driver not existing with said judgment procedure.

[Claim 14] The information processing approach according to claim 13 characterized by performing the directions procedure of directing activation of deletion of a driver, in said driver deletion procedure.

[Claim 15] The information processing approach according to claim 14 characterized by performing deletion of two or more drivers in said driver deletion procedure according to one directions by said directions procedure.

[Claim 16] The information processing approach according to claim 13 characterized by having the verification procedure which checks to a user whether deletion is performed or not before deleting said driver in said driver deletion procedure.

[Claim 17] The information processing approach according to claim 13 characterized by having the deletion prohibition configuration procedure which performs a prohibition setup of deletion of a desired driver.

[Claim 18] Said information processor is the information processing approach according to claim 10 which communicates with an external information processor, is judged that the peripheral device corresponding to said driver does not exist in said judgment procedure, and is characterized by having further the notice procedure of driver deletion of performing the notice of driver deletion which makes said external information processor deleting a driver when said peripheral device is a local device.

[Claim 19] The control program characterized by performing setup information acquisition processing which is the control program performed in the information processor which communicates with a peripheral device, and acquires the setup situation of the driver in said information processor, and judgment processing which judges whether the peripheral device corresponding to the driver currently set up in said information processor exists in said information processor.

[Claim 20] The control program according to claim 19 characterized by performing said information processor so that it may judge whether a peripheral device exists by whether it asks an output destination change port and there is any response in said judgment processing.

[Claim 21] It is the control program according to claim 19 characterized by connecting said information processor to said peripheral-device list through a network.

[Claim 22] The control program according to claim 19 characterized by making said information processor perform further driver deletion which deletes said driver when judged with the peripheral device corresponding to said driver not existing by said judgment processing.

[Claim 23] The control program according to claim 22 characterized by making said information processor perform further directions processing which directs activation of deletion of a driver in said driver deletion.

[Claim 24] The control program according to claim 23 characterized by performing said information processor in said driver deletion

according to one directions by said directions processing so that two or more drivers may be deleted.

[Claim 25] The control program according to claim 22 characterized by making said information processor perform further check processing which checks to a user whether deletion is performed or not before deleting said driver by said driver deletion.

[Claim 26] The control program according to claim 22 characterized by making said information processor perform further deletion prohibition setting processing in which a prohibition setup of deletion of a desired driver is performed.

[Claim 27] Said information processor is a control program according to claim 19 which communicates with an external information processor, is judged that the peripheral device corresponding to said driver does not exist by said judgment processing, and is characterized by making said information processor perform further notice processing of driver deletion in which the notice of driver deletion which makes said external information processor delete a driver is performed when said peripheral device is a local device.

[Claim 28] The information processor characterized by to equip said external information processor with a notice means of driver deletion perform the notice of driver deletion which makes the driver corresponding to said peripheral device delete when judged with the peripheral device corresponding to said driver not existing with a judgment means to judge whether the peripheral device corresponding to the driver which are an external information processor and the information processor which communicates, and is set up in the peripheral-device list exists, and said judgment means.

[Claim 29] It is the information processor according to claim 28 which is further equipped with a setup information acquisition means to acquire the setup situation of the driver in said external information processor, and is characterized by said notice means of driver deletion performing the notice of deletion of a driver based on said setup information.

[Claim 30] The information processor according to claim 28 characterized by having a directions means to direct activation of the notice of deletion of the driver by said notice means of driver deletion.

[Claim 31] Said notice means of driver deletion is an information processor according to claim 30 characterized by performing the notice of driver deletion which makes said external information processor delete two or more drivers according to one directions by said directions means.

[Claim 32] The information processor according to claim 28 characterized by having a check means to check to a user whether deletion is performed or not before said notice means of driver deletion notified deletion of said driver.

[Claim 33] The information processor according to claim 28 characterized by having a deletion prohibition setting means to perform a prohibition setup of deletion of a desired driver.

[Claim 34] The information processor according to claim 28 which said judgment means judges with the peripheral device corresponding to said driver not existing, and is characterized by performing the notice of driver deletion which makes a driver delete to said external information processor when said peripheral device is a local device.

[Claim 35] The information-processing approach characterized by to have the notice procedure of driver deletion perform the notice of driver deletion which makes said external information processor delete the driver corresponding to said peripheral device when judged with the peripheral device corresponding to said driver not existing with the judgment procedure judge whether the peripheral device corresponding to the driver which is the information-processing approach in an external information processor and the information processor which communicates, and is set up exists in a peripheral-device list, and said judgment procedure.

[Claim 36] It is the information processing approach according to claim 35 which has further the setup information acquisition procedure which acquires the setup situation of the driver in said external information processor, and is characterized by said notice procedure of driver deletion performing the notice of deletion of a driver based on said setup information.

[Claim 37] The information processing approach according to claim 35 characterized by having the directions procedure of directing the notice of deletion of the driver by said notice procedure of driver deletion.

[Claim 38] Said notice procedure of driver deletion is the information processing approach according to claim 37 characterized by performing the notice of driver deletion which makes said external information processor perform deletion of two or more drivers according to one directions by said directions procedure.

[Claim 39] The information processing approach according to claim 35 characterized by having the verification procedure which checks to a user whether said external information processor is made to perform deletion before said notice procedure of driver deletion performs said notice of driver deletion.

[Claim 40] The information processing approach according to claim 35 characterized by having the deletion prohibition configuration procedure which performs a prohibition setup of deletion of a desired driver.

[Claim 41] The information processing approach according to claim 35 which judges with the peripheral device corresponding to said driver not existing in said judgment procedure, and is characterized by performing the notice of driver deletion which makes a driver delete to said external information processor when said peripheral device is a local device.

[Claim 42] The judgment processing which judges whether the peripheral device corresponding to the driver which is the control program performed in an external information processor and the information processor which communicates, and is set up in the peripheral-device list exists, When judged with the peripheral device corresponding to said driver not existing by said judgment processing The control program characterized by making said information processor perform notice processing of driver deletion in which the notice of driver deletion which makes said external information processor delete the driver corresponding to said peripheral device is performed.

[Claim 43] It is the control program according to claim 42 characterized by making said information processor perform setup information acquisition processing which acquires the setup situation of the driver in said external information processor, and performing said information processor so that said notice processing of driver deletion may perform the notice of deletion of a driver based on said setup information.

[Claim 44] The control program according to claim 42 characterized by making said information processor perform further directions processing which directs activation of the notice of deletion of the driver by said notice processing of driver deletion.

[Claim 45] Said notice processing of driver deletion is a control program according to claim 44 characterized by performing said information processor so that the notice of driver deletion which makes two or more drivers delete to said external information processor may be performed according to one directions by said directions processing.

[Claim 46] The control program according to claim 42 characterized by making an information processor perform further check processing which checks to a user whether said external information processor is made to perform deletion before said notice processing of driver deletion performs said notice of driver deletion.

[Claim 47] The control program according to claim 42 characterized by making said information processor perform further deletion prohibition setting processing in which a prohibition setup of deletion of a desired driver is performed.

[Claim 48] The control program according to claim 42 which judges with the peripheral device corresponding to said driver not existing by said judgment processing, and is characterized by performing said information processor so that the notice of driver deletion which makes a driver delete to said external information processor may be performed when said peripheral device is a local

device.

[Claim 49] The information processor characterized by to have a notice means perform the notice of deletion of the driver corresponding to said local device to an external information processor when said local device cannot be used from an external information processor according to the connection situation of the local device recognized by recognition means recognize the connection situation of the local device connected to said information processor at the peripheral-device list in two or more external information processors and the information processor which communicates, and said recognition means.

[Claim 50] Said notice means is an information processor according to claim 49 characterized by performing the notices of deletion of a driver all at once to two or more external information processors.

[Claim 51] Said recognition means is an information processor according to claim 49 characterized by having the function to recognize whether said local device exists, and the function to recognize whether sharing setup of said local device is carried out.

[Claim 52] Said notice means is an information processor according to claim 51 characterized by performing the notice of deletion of said driver when recognized as said local device not existing with said recognition means, or when it has been recognized as sharing setup of said local device not being carried out.

[Claim 53] The information processor according to claim 49 characterized by recognizing said two or more information processors which are carrying out sharing setup of said local device based on the setup information which received from said two or more information processors.

[Claim 54] The information-processing approach characterized by to have the notice procedure perform the notice of the driver corresponding to said local device of deletion to an external information processor when said local device cannot be used from an external information processor according to the connection situation of the local device recognized by the recognition procedure recognize the connection situation of the local device connected to said information processor at the peripheral-device list in the information-processing approach in two or more external information processors and the information processor which communicates, and said recognition procedure.

[Claim 55] Said notice procedure is the information processing approach according to claim 54 characterized by performing the notices of deletion of a driver all at once to two or more external information processors.

[Claim 56] Said recognition procedure is the information processing approach according to claim 54 characterized by having the procedure of recognizing whether said local device existing, and the procedure of recognizing whether sharing setup of said local device being carried out.

[Claim 57] Said notice procedure is the information processing approach according to claim 56 characterized by performing the notice of deletion of said driver when recognized as said local device not existing with said recognition procedure, or when it has been recognized as sharing setup of said local device not being carried out.

[Claim 58] The information processing approach according to claim 54 characterized by recognizing said two or more information processors which are carrying out sharing setup of said local device based on the setup information which received from said two or more information processors.

[Claim 59] The control program characterized by to make said information processor perform the notice processing which performs the notice of the driver corresponding to said local device of deletion to an external information processor when said local device cannot be used from an external information processor according to the connection situation of the local device recognized by the recognition processing which recognizes the connection situation of the local device connected to said information processor at the peripheral-device list in the control program in two or more external information processors and the information processor which communicates, and said recognition processing.

[Claim 60] Said notice processing is a control program according to claim 59 characterized by performing said information processor so that the notices of deletion of a driver may be performed all at once to two or more external information processors.

[Claim 61] Said recognition processing is a control program according to claim 59 characterized by making said information processor perform further processing which recognizes whether said local device exists, and processing which recognizes whether sharing setup of said local device is carried out.

[Claim 62] Said notice processing is a control program according to claim 61 characterized by performing said information processor so that the notice of deletion of said driver may be performed when recognized as said local device not existing by said recognition processing, or when it has been recognized as sharing setup of said local device not being carried out.

[Claim 63] The control program according to claim 59 characterized by performing said information processor so that said two or more information processors which are carrying out sharing setup of said local device may be recognized based on the setup information which received from said two or more information processors.

[Claim 64] The information processor characterized by having an input means to input the notice of deletion of the driver which includes assignment of a driver through an external network, and a driver deletion means to delete the specified driver when said notice of deletion is inputted.

[Claim 65] The information processor according to claim 64 characterized by having a check means to check to a user whether deletion of said driver is performed before deleting said driver with said driver deletion means.

[Claim 66] The information processor according to claim 64 characterized by having a deletion prohibition setting means to set up prohibition of deletion of a desired driver.

[Claim 67] The information processor according to claim 66 characterized by having the control means controlled so that said driver deletion means does not delete a driver in said deletion prohibition setting means when prohibition of driver deletion is set up.

[Claim 68] Said information processor is an information processor according to claim 64 characterized by having communicated with the external information processor and having a collection means to collect setup information.

[Claim 69] Said information processor is an information processor according to claim 67 characterized by answering a demand from an external information processor and returning setup information.

[Claim 70] It is the information processor according to claim 64 which is further equipped with a decision means to judge whether the driver specified by said notice of deletion is set up in the information processor, and is characterized by said driver deletion means disregarding said notice of deletion when the driver specified by said decision means is not set up.

[Claim 71] The information processing approach characterized by having the input procedure which inputs the notice of deletion of the driver which includes assignment of a driver through an external network, and the driver deletion procedure of deleting the specified driver when said notice of deletion is inputted.

[Claim 72] The information processing approach according to claim 71 characterized by having the verification procedure which checks to a user whether deletion of said driver is performed before deleting said driver with said driver deletion procedure.

[Claim 73] The information processing approach according to claim 71 characterized by having the deletion prohibition configuration procedure which sets up prohibition of deletion of a desired driver.

[Claim 74] The information processing approach according to claim 73 characterized by having the control procedure controlled by said driver deletion procedure in said deletion prohibition configuration procedure not to delete a driver when prohibition of driver deletion is set up.

[Claim 75] Said information processor is the information processing approach according to claim 71 characterized by communicating with an external information processor and having the collection procedure of collecting setup information.

[Claim 76] Said information processor is the information processing approach according to claim 75 characterized by answering a demand from an external information processor and returning setup information.

[Claim 77] It is the information processing approach according to claim 71 which has further the decision procedure of judging whether the driver specified by said notice of deletion being set up in the information processor, and is characterized by said driver deletion procedure disregarding said notice of deletion when the driver specified by said decision procedure is not set up.

[Claim 78] The control program characterized by making an information processor perform input process which inputs the notice of deletion of the driver which includes assignment of a driver through an external network, and driver deletion which deletes the specified driver when said notice of deletion is inputted.

[Claim 79] The control program according to claim 78 characterized by making said information processor perform check processing which checks to a user whether deletion of said driver is performed before deleting said driver by said driver deletion.

[Claim 80] The control program according to claim 78 characterized by making said information processor perform deletion prohibition setting processing in which prohibition of deletion of a desired driver is set up.

[Claim 81] The control program according to claim 78 characterized by making said information processor perform control processing controlled by said driver deletion in said deletion prohibition setting processing not to delete a driver when prohibition of driver deletion is set up.

[Claim 82] Said information processor is a control program according to claim 78 characterized by communicating with an external information processor and making said information processor perform collection processing which collects setup information.

[Claim 83] Said information processor is a control program according to claim 82 characterized by answering a demand from an external information processor, and performing said information processor so that setup information may be returned.

[Claim 84] It is the control program according to claim 78 which said information processor is made to perform further the decision procedure of judging whether the driver specified by said notice of deletion being set up in the information processor, and is characterized by said driver deletion performing said information processor so that said notice of deletion may be disregarded, when the driver specified by said decision processing is not set up.

[Claim 85] A judgment means to be the network system which contains the second information processor in a peripheral-device list at the first information-processor list, and to judge whether the peripheral device corresponding to the driver currently set up in said first information processor exists, A notice means of driver deletion to perform the notice of driver deletion which makes said second information processor delete the driver corresponding to said peripheral device when judged with the peripheral device corresponding to said driver not existing with said judgment means, The network system characterized by having an input means to input the notice of deletion of said driver into said second information processor through a network, and a driver deletion means to delete the specified driver when said notice of deletion is inputted.

[Claim 86] Claims 19-27, 42-48, 59-63, the storage that is characterized by storing the control program of a publication in any 1 term of 78-84 and in which computer reading is possible.

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to a network system, an information processor, an approach, and a control program, and relates to what performs a display, management, etc. of the information processor and peripheral device which were shared on the network.

[0002]

[Description of the Prior Art] Networks, such as LAN, have also spread in recent years with the spread of peripheral devices, such as a personal computer (PC is called hereafter), and a printer, a scanner, a digital camera, and the needs which share a printer, a modem, a scanner, etc. with two or more PCs on a network have also increased. Moreover, many drivers for using these peripheral devices were also offered, and the install approach has also been offered.

[0003]

[Problem(s) to be Solved by the Invention] However, when a new peripheral device was bought by the performance problem or PC connected when using it was changed, it became [that the driver of these peripheral devices is deleted with as, and] in many cases. Therefore, when actually using a peripheral device, there was a problem that will become, or which driver is effective might stop turning out, and working efficiency will worsen that it is hard to choose since it is shown a list table to an unnecessary driver.

[effective]

[0004] It aims at enabling it to delete the driver which became unnecessary [this invention] when it becomes impossible to use sharing setup, deleting the peripheral device currently shared on the network, or being canceled [being made in view of the above points,] by easy actuation.

[0005]

[Means for Solving the Problem] When the information processor of this invention is described as above-mentioned The means for solving a technical problem, the information processor of this invention is an information processor which communicates with a peripheral device, and it has the description at the point equipped with a setup information acquisition means to acquire the setup situation of the driver in said information processor, and a judgment means to judge whether the peripheral device corresponding to the driver currently set up in said information processor exists.

[0006] Moreover, other information processors of this invention have the description at the point which equipped said external information processor with a notice means of driver deletion perform the notice of driver deletion which makes the driver corresponding to said peripheral device delete, when judged with the peripheral device corresponding to said driver not existing with a judgment means judge whether the peripheral device corresponding to the driver which is the information processor which communicates with an external information processor, and is set up in the peripheral-device list exists, and said judgment means.

[0007] Moreover, other information processors of this invention are set at a peripheral-device list to two or more external information processors and the information processor which communicates. According to the connection situation of the local device recognized by recognition means to recognize the connection situation of the local device connected to said information processor, and said recognition means When said local device cannot be used from an external information processor, it has the description at the point equipped with a notice means to perform the notice of deletion of the driver corresponding to said local device to an external information processor.

[0008] Moreover, other information processors of this invention have the description at the point equipped with an input means to input the notice of deletion of the driver which includes assignment of a driver through an external network, and a driver deletion means to delete the specified driver when said notice of deletion is inputted.

[0009]

[Embodiment of the Invention] Hereafter, with reference to a drawing, the gestalt of operation of the network system of this invention, an information processor, an approach, and a control program is explained.

[0010] Drawing 1 is a block diagram explaining the configuration of the information processor of the gestalt of this operation. In this drawing, 1 is a system bus and each configuration block described below through the system bus 1 concerned is connected.

[0011] 2 is CPU (Central Processing Unit). 3 is program memory (PMEM is called hereafter), carries out the program for the processing mentioned later selection/read in from a hard disk 10 suitably, and performs it by CPU2. Moreover, the data inputted from the keyboard 12 are stored in PMEM which is also text memory as code information.

[0012] 4 is the communications control section and controls the I / O data in the communication link port 5. The outputted signal is told to the communication link port of other equipments on a network (a sign 7 is attached in drawing 1) via a communication line 6 from the communication link port 5. The exchange with a printer, a scanner, etc. which are shared on the network is performed through this communications control section 4. Although the gestalt of this operation describes networks, such as LAN, even if the communication link port 5 and communication line 6 which are connected to this communications control section 4 are a general public line, it cannot be overemphasized that this invention is adapted.

[0013] 8 is an external storage control section. 9 and 10 are the disks for data files, for example, 9 is a flexible disk FD and 10 is a hard disk HD.

[0014] 11 is the input-control section and the input unit of a keyboard 12 and mouse 13 grade is connected. A user can perform the operating command of a system etc. by operating a keyboard 12. Moreover, a mouse 13 functions as a pointing device PD for carrying out the processing directions of the image information on CRT16, this moves the cursor on CRT16 to X and the direction arbitration of Y, choose the command icon on a command menu, and processing is directed, and also directions for edit, directions of a drawing location, etc. can be performed.

[0015] 14 is video image memory (VRAM), 15 is the display output-control section, and 16 is CRT (Cathode-Ray Tube). The data displayed on CRT16 are developed as bit map data on VRAM14.

[0016] 17 is a printer control section and performs the output control of data to the printer 18 connected. 1A is an image reader control section, and controls image reader machine 1B connected.

[0017] In addition, the program memorized to PMEM3 with the gestalt of this operation may be memorized by the hard disk 10 and the storage of flexible disk 9 grade. Furthermore, you may memorize on other equipments connected to the network.

[0018] Hereafter, the case where the driver of a peripheral device is set up is explained with reference to drawing 2 -9. Drawing 2 is a flow chart which shows the processing actuation which sets up a driver. First, at step S201, all PCs on a network and the connection status information of a peripheral device are acquired, and the information is memorized.

[0019] Next, the setup status information of the driver of the peripheral device of each PC is acquired at step S202. The drivers of a peripheral device are drivers, such as a printer, a scanner, a digital camera, and FAX. It is possible to collect information by starting the module which acquires the driver information installed in the self-opportunity for example, on each PC as the acquisition approach of setup status information, and to communicate those information through a network.

[0020] Drawing 3 is drawing showing an example of the driver information-structure object currently set up. It consists of PC name, an IP address, OS class, a user name, a number of device drivers, and information on each device driver. Each device driver information consists of device classes, such as a printer and a scanner, a driver name, version information, an output port, and share information. For example, printer 2000 and a version are served as to 1.00.00 by a local and share information, and, as for the printer registered into the 1st, the output port serves as [the driver name] sharing setup (ON).

[0021] Drawing 4 is drawing showing an example of the screen which displays PC and the peripheral device on a network. 401 is the main window as which the icon to which a menu and 402 express a tool bar and 403 expresses PC and the peripheral device on a network is displayed.

[0022] 402a-402g are the icons showing each function performed by operating PC and a peripheral device in a tool bar 402. For example, 402a is an icon for performing the copy function outputted to the printer which image data was read [printer] from the selected scanner and had this image data chosen. It is an icon for performing the function for a FAX function and 402c to read an image data reading function, and for 402b to read 402d of image data, and to perform OCR processing, the function in which 402e manages FAX received data and distribution data, the function to update information 402f, and the function that stops 402g of update processes.

[0023] 403a-403ae is an icon showing PC and the peripheral device which are shared on the network. An icon is changed by the statuses, such as "error generating" etc. PC, a printer, a scanner, the device class of a FAX modem, or "during processing", and these icon 403a-403ae(s) are displayed. The icon to which 403a expresses a self-opportunity, and 403b are the icons showing the domain where the self-opportunity logs on. Since a self-opportunity is a special device, it is displayed in distinction from other PCs.

[0024] Moreover, although shared on the network like icon 403ad, as for PC and the peripheral device with which the driver is not installed, a gray indication of the icon is given. Moreover, the "-" mark is displayed, when not being developed and displayed and the "+" mark is developed like icon 403ab, although there is a connection device like Icons 403d and 403e. A mark is not added when anything does not have a connection device like Icons 403c and 403g.

[0025] Thus, to all PCs on a network and the connection condition of a peripheral device, and a pan, the status can be checked on this screen. In this example, on account of a screen, although all icons are not displayed, all PCs and peripheral devices can be checked using the scroll bar arranged beside the screen.

[0026] Drawing 5 is drawing showing an example of the screen which displays the driver setup situation of each PC on a network. 501 is the main window as which the icon to which a menu and 502 express a tool bar and 503 expresses PC and the peripheral device on a network is displayed.

[0027] It is an icon showing each function performed by operating PC and a peripheral device in a tool bar 502 502a-502g, and the contents are the same as that of the icons 402a-402g shown in drawing 4. 503a-503n are the icons showing PC and the peripheral device which are shared on the network. Here, the driver information on a peripheral device is displayed and it can check.

[0028] If explanation is returned to the flow chart of drawing 2, it will judge whether the driver of a peripheral device is set up at step S203. For example, printer 403ad can be chosen in drawing 4, and setup instruction of a driver can be performed from a menu. Here, this processing is ended when there is no setup instruction of a driver.

[0029] When the setup instruction of a driver occurs, it shifts to step S204 and the driver to set up is directed. Drawing 6 is an example of the screen for choosing the driver to set up, chooses the manufacturer here and directs an object printer. Moreover, a user can also direct a folder with setup information.

[0030] Furthermore, PC which sets up a driver is chosen at step S205. Drawing 7 is an example of the screen for choosing PC which sets up a driver, and chooses a domain and PC here. Under the present circumstances, two or more PCs can be chosen.

[0031] And if there are directions of O.K., it will judge whether it is necessary to set up in PC to which it shifted to step S206 and the directed driver was directed from the driver information from each PC. Install processing is not performed when [- the directed driver is already installed -] it is not necessary to set up by carrying out.

[0032] When it is necessary to set up, it shifts to step S207, and directions are sent so that setup processing of the driver directed at step S204 to PC directed at step S205 may be performed. And setup processing of a driver is performed according to the directions.

Drawing 8 is an example of the driver setup instruction structure, and consists of a device class, a driver name, version information, an output port, setup information, etc. In addition, although setup information may be sent with this structure, it is memorized to the shared folder on a network, and you may make it direct that pathname.

[0033] Next, it judges whether setup processing of a driver terminated normally with step S208. It judges by whether normal actuation is actually carried out using the driver at the same time it judges whether install processing of a driver was performed normally as the judgment approach. For example, when a driver is able to install normally, it enables it to check on a screen, as the message terminated normally to PC which carried out setup instruction is transmitted.

[0034] You may enable it to judge whether by performing a test print further, in the case of the printer, the printing result was seen, and it has installed in it normally. Drawing 9 shows an example of setup confirm printing of a driver. Thus, it can be checked by printing an outputting agency PC name, a driver name, version information, etc. on which PC setup processing has been completed normally. Processing is ended when it terminates normally.

[0035] When it does not terminate normally, it judges whether it shifts to step S209 and activation directions are again carried out to it to PC which was not terminated normally. Processing is ended, when there are directions of rerun and there are no directions of return and rerun in step S204.

[0036] While being able to install a driver by easy actuation when the driver of a peripheral device is not installed in the screen on which PC and the peripheral device which are shared on the network are displayed as stated above, it can check that it has actually

been installed and the working efficiency on a network can be raised sharply.

[0037] Hereafter, with reference to drawing 10 -19, the check of the driver of a peripheral device and the case where it deletes further are explained. Drawing 10 is a flow chart which shows the check of a driver, and the processing actuation which deletes further. First, at step S1001, all PCs on a network and the connection status information of a peripheral device are acquired, and the information is memorized.

[0038] Next, the setup status information of the driver of the peripheral device installed in the self-opportunity is acquired at step S1002. The drivers of a peripheral device are drivers, such as a printer, a scanner, a digital camera, and FAX.

[0039] Drawing 11 is drawing showing an example of the driver information-structure object currently set up. It consists of PC name, an IP address, OS class, a user name, a number of device drivers, and information on each device driver. Each device driver information consists of device classes, such as a printer and a scanner, a driver name, version information, an output port, and share information. For example, LASER-830 and a version are served as to 1.00.00 by local connection and share information, and, as for the printer registered into the 1st, the output port serves as [the driver name] sharing setup (ON).

[0040] Drawing 12 is drawing showing an example of the screen which displayed PC and the peripheral device on a network. 301 is the main window as which the icon to which a menu and 302 express a tool bar and 303 expresses PC and a peripheral device is displayed.

[0041] 302a-302i are the icons showing each function performed by operating PC and a peripheral device in a tool bar 302. For example, 302a is an icon for performing the copy function outputted to the printer which had reading of image data and this image data chosen by the selected scanner. It is an icon for performing the function for a FAX function and 302c to read an image data reading function, and for 302b to read 302d of image data, and to perform OCR processing, the function in which 302e manages FAX received data and distribution data, the function in which perform a display change function and 302g of display screen edit functions, and 302h updates information 302f, and the function in which 302i stops an update process.

[0042] 303a-303q are the icons showing PC and the peripheral device which are shared on the network. An icon is changed by the statuses, such as "error generating" etc. PC, a printer, a scanner, the device class of a FAX modem, or "during processing", and these icons 303a-303q are displayed. The icon to which 303c expresses a self-opportunity, and 303b are the icons showing the domain where the self-opportunity logs on. Since a self-opportunity is a special device, it is displayed at a head in distinction from other PCs. About other PCs, it is displayed on an alphabetical order in ascending order or descending order.

[0043] Moreover, although shared on the network like icon 303p, as for PC and the peripheral device with which the driver is not installed, a gray indication of the icon is given. Moreover, the "-" mark is displayed, when not being developed and displayed and the "+" mark is developed like Icons 303h, 303k, and 303n, although there is a connection device like icon 303j. A mark is not added when anything does not have a connection device like icon 303q.

[0044] Thus, to all PCs on a network and the connection condition of a peripheral device, and a pan, the status can be checked on this screen. In this example, on account of a screen, although all icons are not displayed, all PCs and peripheral devices can be checked using the scroll bar arranged beside the screen.

[0045] Drawing 13 is drawing showing an example of the screen displayed when it connects with PC expressed with icon 303h, and a printer-sharing setup expressed with icon 303i by which sharing setup was carried out on the network is canceled or a driver is deleted from the condition shown in drawing 12. As shown in this drawing, the display "x" mark which shows "use is improper" is displayed on icon 303i. In this condition, although the driver is installed, since an actual output destination change does not exist, it cannot output.

[0046] If explanation is returned to the flow chart of drawing 10, it will judge whether the driver of a peripheral device is checked at step S1003. For example, check directions of a driver can be performed by a screen as shown in drawing 14 being displayed, and choosing the device to check. On this screen, since there is no device when it checks, when it is judged with it being unnecessary, a setup of whether to carry out a message indicator can also be specified collectively. Here, this processing is ended when there are no check directions of a driver.

[0047] When there are check directions of a driver, it shifts to step S1004 and judges whether all the drivers made applicable to a check were checked. This processing is ended when it checks about all drivers.

[0048] When all drivers are not being checked, it judges whether it shifts to step S1005 and a deletion prohibition setup of the driver to check is carried out. For example, when check processing is performed by the printer usually used well, in it being judged with there being no object device, and being deleted, since it was carrying out [be / by chance / the power source / turned on], a driver must be reinstalled again and it is inconvenient to a user.

[0049] Then, it enables it to perform a deletion prohibition setup to a driver not to delete. Drawing 15 is an example of the screen for performing a deletion prohibition setup of a driver, a driver to forbid deletion here can be chosen and a deletion prohibition setup can be performed on a prohibition list by carrying out additional registration. On the contrary, the driver which wants to cancel the ban on deletion can be chosen, and it can also delete from a prohibition list.

[0050] At the above-mentioned step S1005, when forbidden, return and the following driver are checked to step S1004. When not forbidden, it shifts to step S1006 and judges whether the device corresponding to a driver exists. For example, if it is a printer, it will judge by whether it asks the output port point and there is any response. To a scanner etc., it checks by performing same processing to input port. When an object device exists, return and the following driver are checked to step S1004.

[0051] When an object device does not exist, it shifts to step S1007 and judges whether a deletion message is displayed. This judgment is performed according to the information set up at step S1003. Here, when being set up so that a message may be displayed, it shifts to step S1008. When being set up so that a message may not be displayed, it shifts to step S1009.

[0052] The message to which assignment of whether to perform deletion to a user is urged is expressed as step S1008. Drawing 16 is an example of the message displayed when there is a driver to delete. Thus, it directs whether an object driver displayed and delete. Here, when "yes" is directed, it shifts to step S1009. When "no" is directed, deletion is canceled and return and the following driver are checked to step S1004.

[0053] Deletion of the specified driver is performed at step S1009. And after processing finishes, return and the following driver are checked to step S1004.

[0054] Although the driver is checked by computer by which sharing setup is carried out in the device in the flow chart shown in drawing 10, it thinks, also when the computer which is carrying out sharing setup of the device performs deletion directions of a driver. Drawing 17 and 18 show a flow chart in case the computer (information processor of this invention) which is carrying out sharing setup of the device performs deletion directions of a driver.

[0055] First, steps S801 and S802 are the same processings as steps S1001 and S1002. It judges whether at step S803, all were confirmed about the driver installed. This processing is ended when it checks about all drivers.

[0056] When all drivers are not being checked, it shifts to step S804 and the device corresponding to a driver judges a local device and

the device shared between the network. When it is not a local device, return and the following driver are checked to step S803.

[0057] When it is a local device, it shifts to step S805 and judges whether the device corresponding to a driver exists. The judgment approach is the same as that of step S1005. When the device corresponding to a driver exists, it shifts to step S811.

[0058] When the device corresponding to a driver does not exist, it shifts to step S806 and judges whether the driver was performing sharing setup. When sharing setup has not been carried out, it shifts to step S808.

[0059] When sharing setup is being carried out, it shifts to step S807, and it notifies to other computers that it became impossible to use an object device through a network, and shifts to step S808. Drawing 19 is an example of the information-structure object which notifies deletion of a driver, and has the information on a device class, a driver name, version information, an output port, etc.

[0060] In the above-mentioned processing, the setup status information of a driver is acquired from the peripheral device of each PC (external information processor) by computer (information processor of this invention) which is carrying out sharing setup of the device. The drivers of a peripheral device are drivers, such as a printer, a scanner, a digital camera, and FAX. It is possible that information is required and collected on each PC as the acquisition approach of setup status information by starting the module which acquires the driver information installed in the self-opportunity, for example, and this information processor communicates those information through a network from an external information processor (each PC).

[0061] Drawing 21 is drawing showing an example of the driver information-structure object currently set up. It consists of PC name, an IP address, OS class, a user name, a number of device drivers, and information on each device driver. Each device driver information consists of device classes, such as a printer and a scanner, a driver name, version information, an output port, a shared name, and the driver information address. for example, the printer registered into the 1st -- 1.00.00 and an output port become [LASER-830 and the driver information address] a local, and the shared name has become [the driver name / LASER-830 and a version] Ox10000. This information is recorded for every PC.

[0062] By transmitting and receiving setup information between each PC as mentioned above, the driver of the device by which sharing setup is carried out in each PC can be recognized. PC (information processor of this invention) which is carrying out sharing setup of the local device it became impossible to use other PCs notifies that it became impossible to use the local device concerned to PC which has the driver of the local device concerned by which sharing setup is carried out using this information. If each PC which received the notice concerned receives the notice including assignment of the driver corresponding to the device concerned, as shown in drawing 18, deletion of the driver corresponding to the device concerned will be performed. Moreover, before transmitting to PC to which deletion of a driver is notified from PC which is carrying out sharing setup of the local device for which it became impossible for other PCs to use the notice of deletion of a driver In PC which is carrying out sharing setup of the local device it became impossible to use PC being concerned -- others -- the directions input screen (directions means) which carries out the directions input of whether transmitting processing of the notice of driver deletion is made to carry out to OS -- or In PC of the notice place of the notice of driver deletion, the directions screen where a user can direct activation of transmitting processing of the notice of deletion in the message of whether to make a driver delete and a list may be displayed. Moreover, you may make it make two or more drivers delete to PC of a notice place by the notice of one driver deletion.

[0063] In addition, although PC which is setting up the driver of the target local device was distinguished and the deletion directions by shared discharge were performed only to the PC, when it does not restrict to this, it performs deletion directions to all PCs on a network and the driver is set up in person in each PC, you may make it make deletion perform with the gestalt of this operation.

[0064] At step S808, it judges whether a deletion prohibition setup of the object driver is carried out. The judgment approach is the same as that of the case of step S1005. When a deletion prohibition setup is carried out, return and the following driver are checked to step S803.

[0065] When a deletion prohibition setup is not carried out, it shifts to step S809 and judges whether deletion is performed or not. The decision approach is the same as that of step S1008. Here, when deletion of a driver is canceled, return and the following driver are checked to step S803.

[0066] When deletion activation of a driver is directed, it shifts to step S810 and deletion of a driver is performed, and after ending deletion, return and the following driver are checked to step S803.

[0067] At step S811, it judges whether the object driver canceled sharing setup. Here, when the object driver is not having sharing setup canceled, return and the following driver are checked to step S803.

[0068] When the object driver has canceled sharing setup, it shifts to step S812 and notifies to other computers that it became impossible to use an object device through a network.

[0069] By computer it is notified on the other hand that a notice of driver deletion which was mentioned above is, as shown in drawing 18, it judges first whether the notice of deletion was received at step S901. Processing is ended when having not received the notice.

[0070] When a notice is received, it shifts to step S902 and judges whether there is any driver corresponding to the notice of driver deletion. Processing is ended when there is no corresponding driver.

[0071] When there is a corresponding driver, it shifts to step S903 and judges whether a deletion prohibition setup of the object driver is carried out. The judgment approach is the same as that of step S1005. Processing is ended when a deletion prohibition setup of the driver is carried out.

[0072] When a deletion prohibition setup is not carried out, it shifts to step S904 and judges whether a message is displayed or not. When the message indicator setup is not carried out, it shifts to step S906. When a message indicator setup is carried out, it shifts to step S905 and the acknowledgement message of whether to delete is displayed.

[0073] And processing is ended when the input directions of the deletion directions are not carried out at step S905. When the input directions of the deletion directions of a driver are carried out, it shifts to step S906, an object driver is deleted, and processing is ended.

[0074] When it becomes impossible for the peripheral device currently shared on the network to use it in the screen on which PC and the peripheral device which are shared on the network are displayed, having carried out [that deletion or sharing setup is canceled etc. and] as stated above, the driver which became unnecessary can be uninstalled by easy actuation. Moreover, the driver of the peripheral device which was carrying out sharing setup can be deleted, or when sharing setup is canceled, an unnecessary driver can be deleted by notifying to other computers. By these functions, the working efficiency on a network can be raised sharply.

[0075] Hereafter, the case where the driver of a peripheral device is upgraded is explained with reference to drawing 20 -31. Drawing 20 is a flow chart which shows the processing actuation which upgrades a driver. First, at step S2001, all PCs on a network and the connection status information of a peripheral device are acquired, and the information is memorized.

[0076] Next, setup status information is acquired to the driver of the peripheral device of each PC at step S2002. The drivers of a peripheral device are drivers, such as a printer, a scanner, a digital camera, and FAX. It is possible to collect information by starting the module which acquires the driver information installed in the self-opportunity on each PC as the acquisition approach of setup status information, for example, and to communicate those information through a network.

[0077] Drawing 21 is drawing showing an example of the driver information-structure object currently set up. It consists of PC name, an IP address, OS class, a user name, a number of device drivers, and information on each device driver. Each device driver information consists of device classes, such as a printer and a scanner, a driver name, version information, an output port, a shared name, and the driver information address. For example, the printer registered into the 1st -- 1.00.00 and an output port become [LASER-830 and the driver information address] a local, and the shared name has become [the driver name / LASER-830 and a version] Ox10000. This information is recorded for every PC.

[0078] All PCs on a network and the connection status information of a peripheral device are expressed as step S2003 based on the acquired information. Since it mentioned already by drawing 12 about this screen display, that detailed explanation is omitted.

[0079] Drawing 22 is drawing showing an example of the screen which displayed the driver setup situation of each PC on a network, and is displayed based on the driver information-structure object acquired from each PC. For example, as for the driver installed in my PC, in this drawing, a printer has six pieces and two scanners. In this, it is shown that a driver name is "INKJET-10V", a version number is "1.00.00", and the printer "INKJET-10V" is a printer driver currently shared on the network where an output port name is "\\NOTPC01\INKJET-10V", and sharing setup is carried out by the name "INKJET-10V."

[0080] If explanation is returned to the flow chart of drawing 20, it will judge whether the driver of a peripheral device was updated at step S2004. Here, processing is ended when the driver is not updated. When the driver is updated, it shifts to step S2005. As the updating approach of a driver, a driver is updated on each PC and it is possible to register the driver information into PC used as a server machine. Or it is also considered that a user downloads and registers the newest driver from each manufacturer's site.

Furthermore, the specific location on a network is specified and the version check of a driver can also be performed. Drawing 23 is an example of the screen for performing a version check setup of a driver. Thus, when the version check of a driver is performed periodically and there is renewal of a driver by setting up specific URL, it can set up so that it may acquire automatically.

[0081] At step S2005, it judges whether the updated driver is the latest version. And when it is not the latest version, since it is not necessary to give the notice to update, processing is ended. When it is the latest version, it shifts to step S2006. It is possible to carry out as the judgment approach of the version of a driver based on the driver information on each printer managed with the server vessel. Drawing 24 is an example of the structure by which version control is carried out for every driver, and consists of device classes, such as a printer and a scanner, the driver name of a device, a version information number managed, driver version information, and driver information corresponding to it. It judges by comparing the managed version information with the version information of the updated driver.

[0082] At step S2006, it judges whether there is a PC which is using the updated driver. A judgment is performed using the device driver information-structure object acquired from each PC. Here, processing is ended when there is no object PC. When there is an object PC, an object PC list is displayed. Drawing 25 is an example of the screen for choosing the candidate PC for updating of a driver, if Object PC is chosen here and "O.K." is chosen, it will shift to step S2007 and a notice will be performed to selected PC. Moreover, selection of "choosing all the objects PC" chooses all the objects PC.

[0083] At step S2007, it notifies that the driver was updated to the selected object PC. Drawing 26 is an example of the information-structure object of the notice of renewal of a device, and consists of a device class, a driver name, version information, and an output port.

[0084] And it judges whether it shifts to step S2008 and there is any acquisition demand of driver information from notified PC. When there is no demand, or when driver information did not need to be transmitted and there is a response from Client PC, it shifts to step S2010. When there is a demand, it shifts to step S2009 and driver information with a demand is transmitted to Client PC.

[0085] At step S2010, it judges whether there was any response from all notified PCs. Processing is ended when there is a response from all PCs. When there is still no response, it returns from all PCs to step S2008.

[0086] On the other hand, drawing 27 is a flow chart which shows the processing actuation in a client device. First, it judges whether there was any notice of updating of a driver from a server machine at step S901. Here, processing is ended when there is no notice.

[0087] When there is a notice, it shifts to step S902 and judges whether renewal of a driver is carried out. For example, the approach which carries out the message indicator of there having been a notice of renewal of a driver, and a user directs can be considered.

Drawing 28 is an example of the message which notifies that the driver was updated. Here, when "no" is chosen, it notifies that object driver information is not required to a server machine, and processing is ended. When "yes" is chosen, it shifts to step S903 and object driver information is required from a server machine.

[0088] Next, it judges whether driver information was acquirable at step S904. When it cannot acquire, an acquisition demand is again transmitted to step S903 to return and a server machine. When driver information is acquirable, it shifts to step S905 and a driver is updated. Furthermore, the driver install situation newest at step S906 is notified to a server machine, and processing is ended.

[0089] In the flow chart shown in drawing 20 and 27, when an update process of a driver is carried out, PC used as a server machine is performing selection of PC which notifies, but also when judging from the driver situation installed in the self-opportunity with PC used as a client device and updating a driver, it thinks. Drawing 29 - 31 are a flow chart in the case of judging from the driver situation installed in the self-opportunity with PC used as a client device, and updating a driver.

[0090] Step S1201 of drawing 29 - step S1205 are the same as step S2001 - step S2005. Next, at step S1206, it notifies to all PCs that the driver was updated, and processing is ended. It is as the information-structure object to notify also having been shown in drawing 26.

[0091] And the demand from a client device is processed according to the flow chart shown in drawing 30. First, it judges whether there is any demand of driver information at step S1301. Processing is ended when there is no demand.

[0092] When there is a demand, it shifts to step S1302 and driver information with a demand is read from the managed driver information. And the driver information is transmitted to a client device with a demand at step S1303.

[0093] On the other hand, in a client device, processing is performed according to the flow chart shown in drawing 31. Although this flow chart is almost the same as the flow chart shown in drawing 27 and step S1401, step S1403 - step S1407 are as step S901 of drawing 27 - step S906 having explained, that there is a judgment of step S1402 differ.

[0094] At this step S1402, when the notice of renewal of a driver is received from a server machine, it judges whether there is any notified object driver as compared with the driver information which reads a data requirement from the notice structure of renewal of a driver, and is installed in the self-opportunity. And when there is no object driver, this notice is disregarded and processing is ended. When there is an object driver, the same processing as the flow chart shown in drawing 27 is performed.

[0095] As stated above, when the driver of a peripheral device is updated in the screen on which PC and the peripheral device which are shared on the network are displayed, a driver can be updated by easy actuation. Moreover, since it can judge whether a driver is updated or not by notifying only the update information of a driver to each PC, the traffic on a network can be stopped to the minimum. By these functions, the working efficiency on a network can be raised sharply. In addition, in each above-mentioned operation gestalt, a driver deletion means may be offered as an uninstallation module of OS. Even in this case, the uninstallation

program (driver deletion means) mounted as application which performs deletion of a driver also serves as criteria of this invention by controlling OS to delete a driver.

[0096] (Gestalt of other operations) So that various kinds of devices may be operated in order to realize the function of the gestalt of operation mentioned above As opposed to the computer in the equipment connected with these various devices, or a system The program code of the software for realizing the function of the gestalt of the above-mentioned implementation is supplied. What was carried out by operating the various above-mentioned devices according to the program stored in the computer (CPU or MPU) of the system or equipment is contained under the category of this invention.

[0097] Moreover, the function of the gestalt of operation which the program code of the above-mentioned software itself mentioned above in this case will be realized, and the record medium which stored the means for supplying that program code itself and its program code to a computer, for example, this program code, constitutes this invention. As a record medium which memorizes this program code, a flexible disk, a hard disk, an optical disk, a magneto-optic disk, CD-ROM, a magnetic tape, the memory card of a non-volatile, ROM, etc. can be used, for example.

[0098] Moreover, by performing the program code with which the computer was supplied, also when the function of the gestalt of above-mentioned operation is not only realized, but the function of the gestalt of above-mentioned operation is realized in collaboration with OS (operating system) or other application software etc. with which the program code is working in a computer, it cannot be overemphasized that this program code is contained in the gestalt of operation of this invention.

[0099] Furthermore, after stored in the memory with which the functional expansion unit by which the supplied program code was connected to the functional add-in board and the computer of a computer is equipped, also when the function of the gestalt of operation which the CPU with which the functional add-in board and functional expansion unit are equipped based on directions of the program code performed a part or all of actual processing, and mentioned above by the processing is realized, it cannot be overemphasized that it is contained in this invention.

[0100] In addition, it passes over no the configurations and structures of each part which were shown in the gestalt of the above-mentioned implementation to what showed a mere example of the somatization which hits carrying out this invention, and the technical range of this invention must not be restrictively interpreted by these. That is, this invention can be carried out in various forms, without deviating from the pneuma or its main description.

[0101]

[Effect of the Invention] When it becomes impossible to use sharing setup according to this invention, deleting the peripheral device currently shared on the network, or being canceled as explained above, the driver which became unnecessary can be uninstalled by easy actuation. Moreover, the driver of the peripheral device which was carrying out sharing setup can be deleted, or an unnecessary driver can be deleted by notifying to other computers, when sharing setup is canceled. By these functions, the working efficiency on a network can be raised sharply.

[Translation done.]

* NOTICES *


JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DRAWINGS

[Drawing 7]

ドライバのセットアップ

 ドライバをセットアップするPCを選択してください。

ドメイン名 (D) :	PC名 (P) :
ドメイン2	SNPC01
	SNPC02
	SNPC03
	SNPC04
	SNPC05
	SNPC06
	SNPC07

[Drawing 8]

デバイス種類	プリンタ
ドライバ名	Printer 2000
バージョン情報	1.00.00
出力ポート	Wsnpcw2VPm2000
セットアップ情報	0

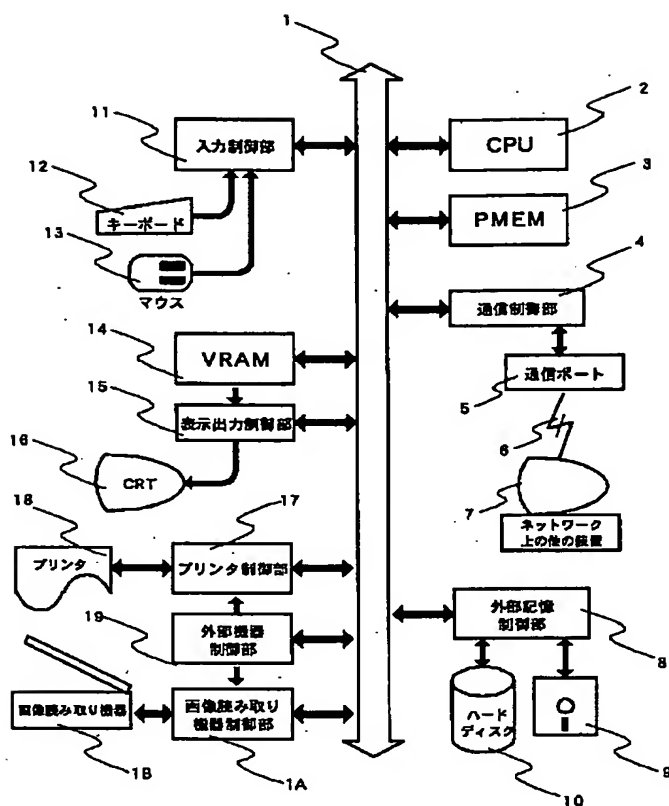
[Drawing 19]

デバイス種類	プリンタ
ドライバ名	Printer 2000
バージョン情報	1.00.00
出力ポート	Wsnpcw2VPm2000

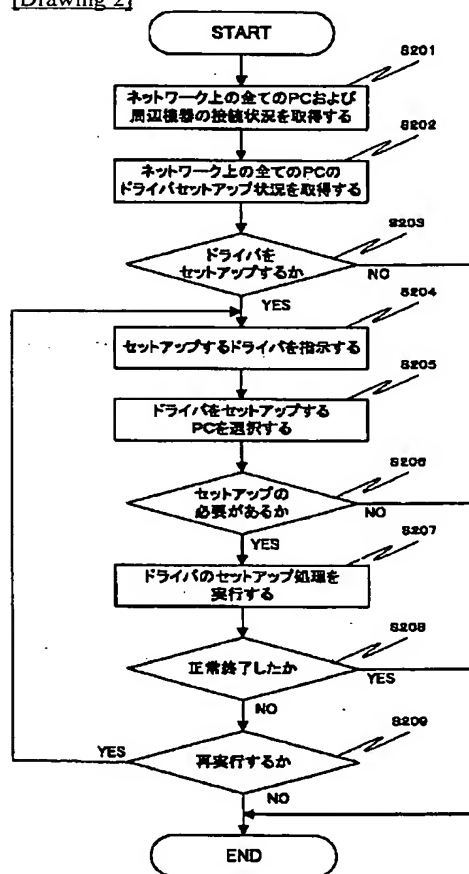
[Drawing 26]

デバイス種類	プリンタ
ドライバ名	Printer 2000
バージョン情報	1.20.00
出力ポート	Wsnpcw2VPm2000

[Drawing 1]



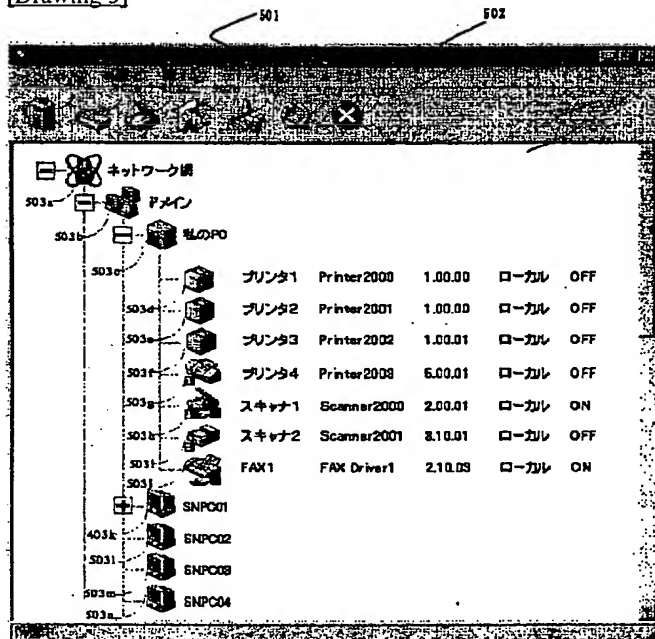
[Drawing 2]



[Drawing 3]

PC名称		SNPC00
IPアドレス		100.100.100.1
OS種類		Windows98
ユーザ名		suzuki
ドライバ数		N
1	デバイス種類	プリンタ
	ドライバ名	Printer 2000
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	ON
2	デバイス種類	スキャナ
	ドライバ名	Scanner 2000
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	ON
		●
		●
		●
		●
		●
N	デバイス種類	プリンタ
	ドライバ名	Printer 2001
	バージョン情報	1.00.00
	出力ポート	ネットワーク
	共有情報	OFF

[Drawing 5]



[Drawing 23]

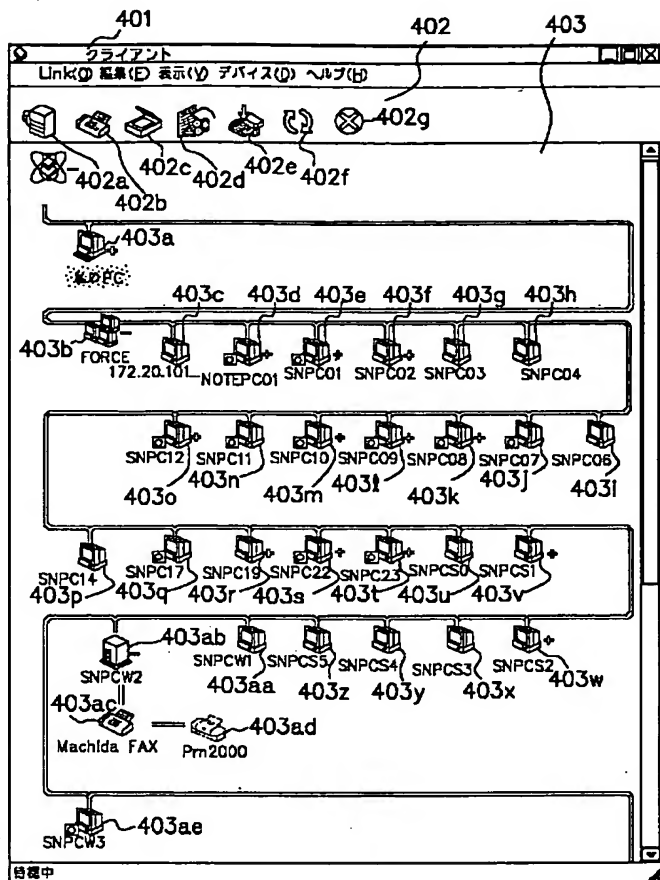
ドライババージョンチェック設定

ドライバのバージョンチェックを行うURLを設定してください。

URL(U):

OK キャンセル

[Drawing 4]



[Drawing 6]

ドライバのセットアップ

プリンタの製造元とモデルを選んでください。プリンタにインストールディスクが付いている場合は、[ディスク使用]を押して下さい。プリンタが一度にない場合は、プリンタのマニュアルを参照して互換性のあるプリンタを選んでください。

製造元 (M):

プリンタ (P):

ディスク使用 (H)

次へ

キャンセル

[Drawing 9]

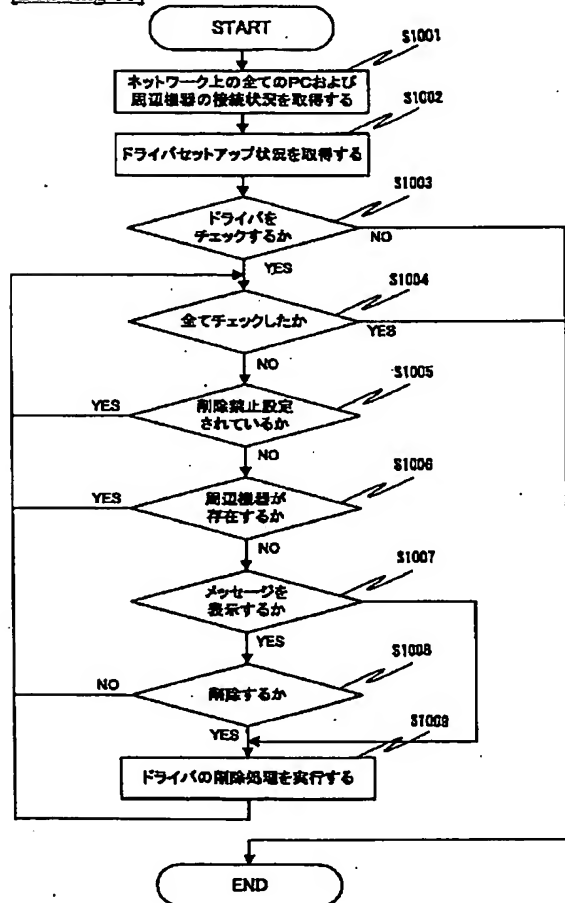
プリンタの印字テスト

Printer 2000 のインストールが完了しました。

PC 名、プリンタドライバおよびポートの設定は以下の通りです。

日時: 2000 年 6 月 12 日
 PC 名: SNPC01
 ドライバ名: Printer2000
 バージョン: 1.00.00
 出力ポート: \\\$SNPCW2\$Pm2000

[Drawing 10]



[Drawing 16]

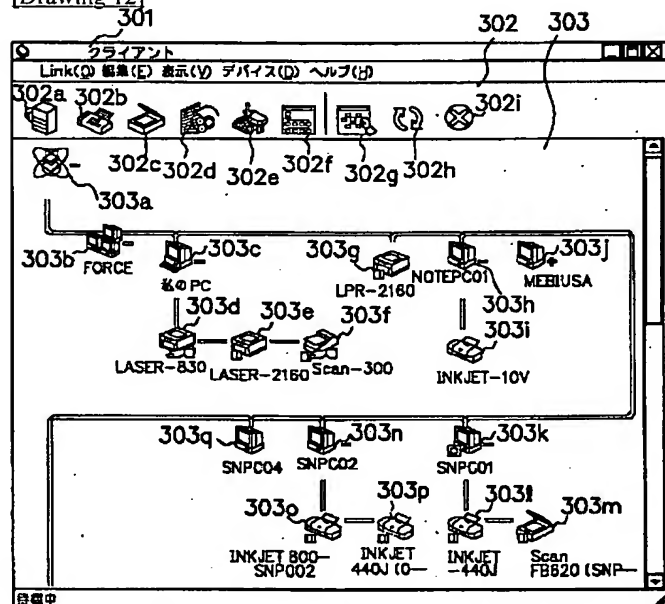
削除確認

? * INKJET-10V* ドライバに対応するプリンタはありません。ドライバを削除しますか?

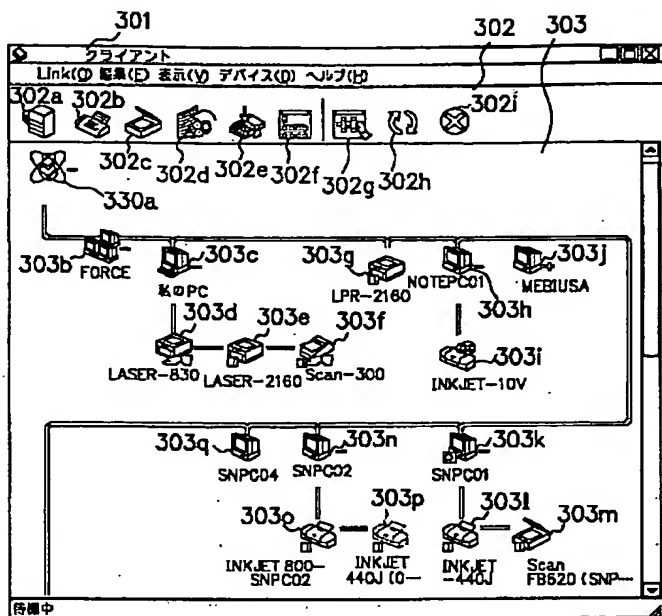
[Drawing 11]

P C 名称		SNPC00
I P アドレス		100.100.100.1
O S 種類		Windows98
ユーザ名		suzuki
ドライバ数		8
1	デバイス種類	プリンタ
	ドライバ名	LASER-830
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	ON
2	デバイス種類	プリンタ
	ドライバ名	LBP-2160
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	OFF
3	デバイス種類	プリンタ
	ドライバ名	Scanner2000
	バージョン情報	1.00.00
	出力ポート	100.100.100.123
	共有情報	OFF
4	デバイス種類	プリンタ
	ドライバ名	INKJET-10V
	バージョン情報	1.00.00
	出力ポート	※※NOTEPC01※INKJET-10V
	共有情報	OFF
7	デバイス種類	スキャナ
	ドライバ名	Scan 300
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	OFF
8	デバイス種類	スキャナ
	ドライバ名	Scan FB620
	バージョン情報	1.00.00
	出力ポート	※※snpc01※FB620
	共有情報	OFF

[Drawing 12]



[Drawing 13]



[Drawing 14]

ドライバのチェック

デバイスの接続状態とインストールされているドライバをチェックして、不要なドライバを削除します。
 実行する場合は、チェックするデバイスを選択して
 [OK] ボタンを選択してください。
 キャンセルする場合は、[キャンセル] ボタンを選択してください。

デバイス種類 (D) :

☐ プリンタ
☐ スキャナ
☐ FAX

☐ 確認メッセージを表示する。

OK キャンセル

[Drawing 15]

ドライバの削除禁止設定

ドライバの削除を自動的に行わないドライバを設定してください。

禁止するドライバ (B)

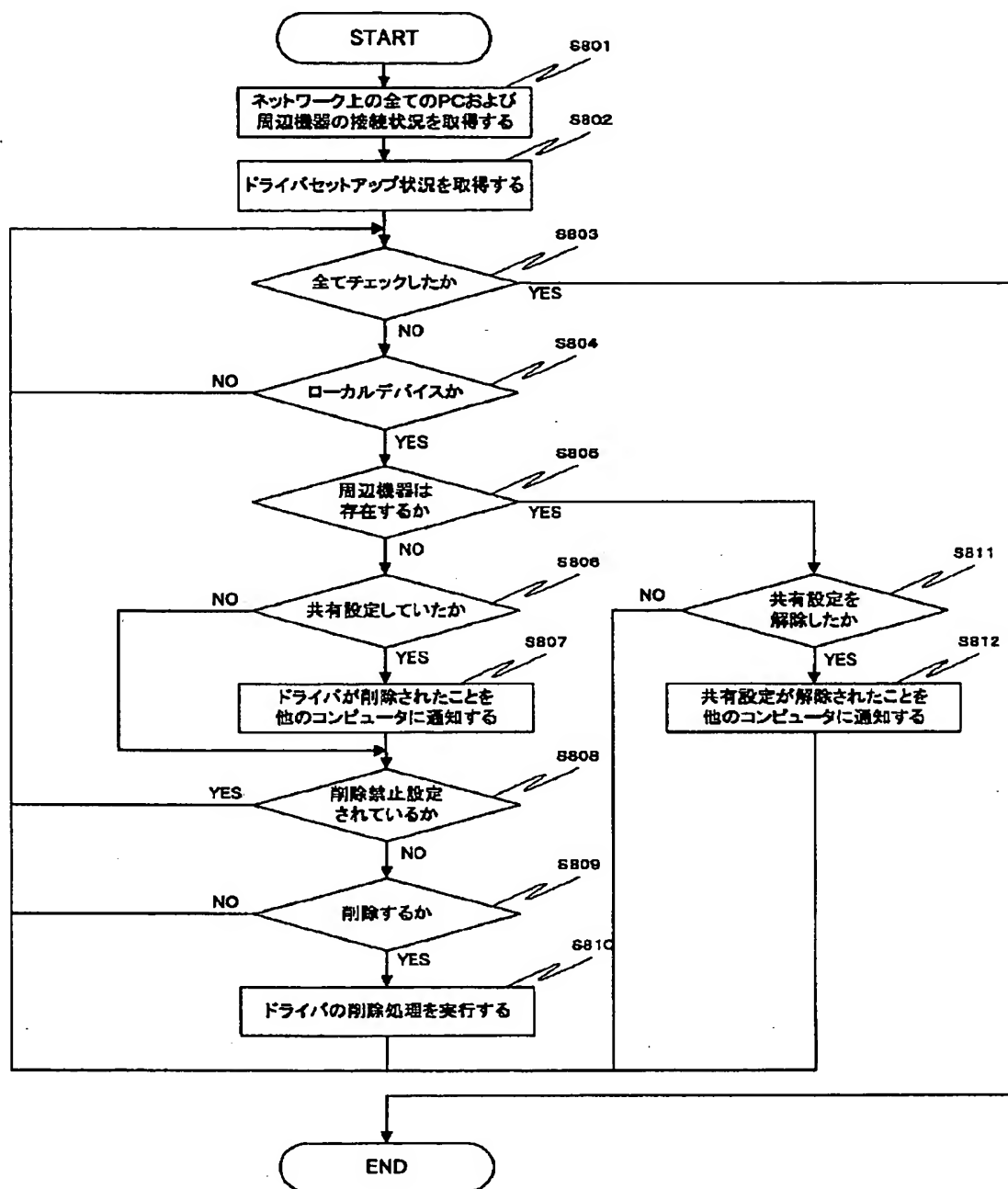
参照 (S) 追加 (A)

禁止するドライバー一覧 (L)

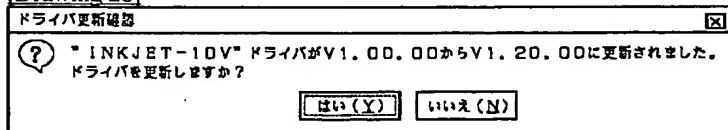
#snpc01 WINKJET-440J 削除 (R)

OK キャンセル

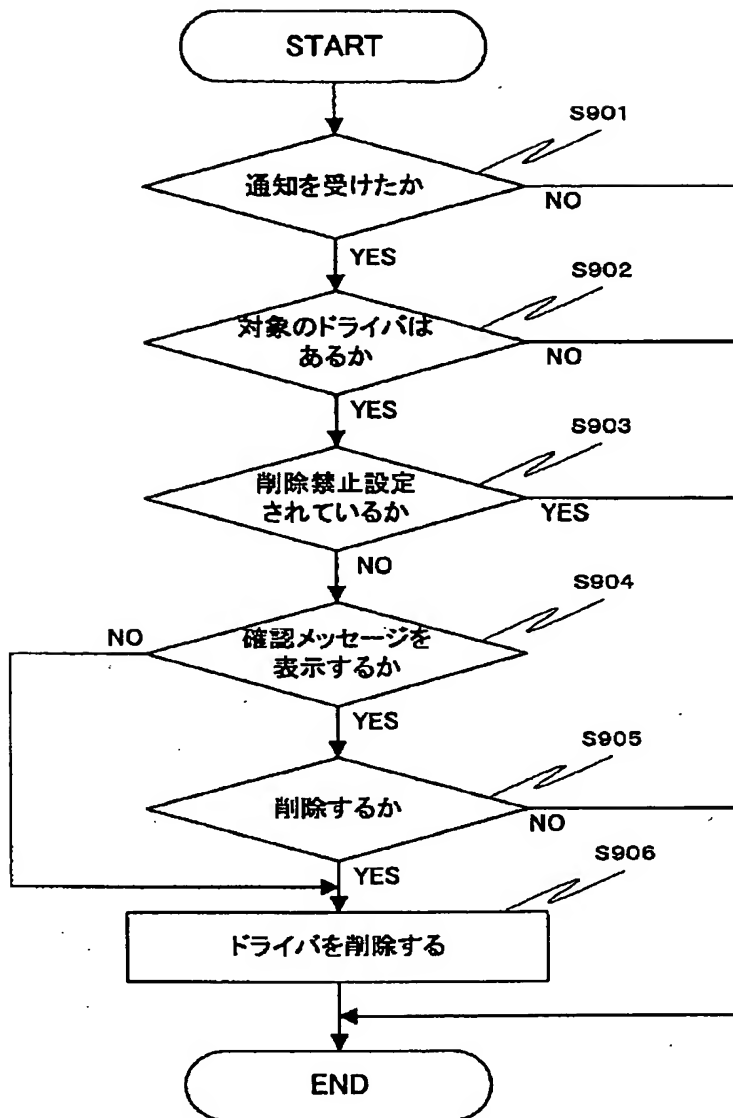
[Drawing 17]



[Drawing 28]



[Drawing 18]



[Drawing 21]


PC名称		SNPC00
IPアドレス		100.100.100.1
OS種類		Windows98
ユーザ名		suzuki
ドライバ数		8
1	デバイス種類	プリンタ
	ドライバ名	LASER-830
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有名	LASER-830
2	ドライバ情報アドレス	0x10000
	デバイス種類	プリンタ
	ドライバ名	LASER-2160
	バージョン情報	1.00.00
	出力ポート	ローカル
3	共有名	0
	ドライバ情報アドレス	0x20000
	デバイス種類	プリンタ
	ドライバ名	INKJET-10V
	バージョン情報	1.00.00
8	出力ポート	***NOTEPC01*INKJET-10V
	共有名	INKJET-10V
	ドライバ情報アドレス	0x30000
8	デバイス種類	スキャナ
	ドライバ名	Scan FB620
	バージョン情報	1.00.00
	出力ポート	***snpc01*FB620
	共有名	FB620
	ドライバ情報アドレス	0x40000
	ドライバ情報	

[Drawing 24]

デバイス種類		プリンタ
ドライバ名		LASER-830
バージョン情報数		N
1	バージョン情報	1.00.00
	ドライバ情報アドレス	0x10000
2	バージョン情報	1.20.00
	ドライバ情報アドレス	0x20000
N	バージョン情報	3.10.00
	ドライバ情報アドレス	0x80000
ドライバ情報		

[Drawing 25]

ドライバのセットアップ

 ドライバ更新通知を行うPCを選択してください。

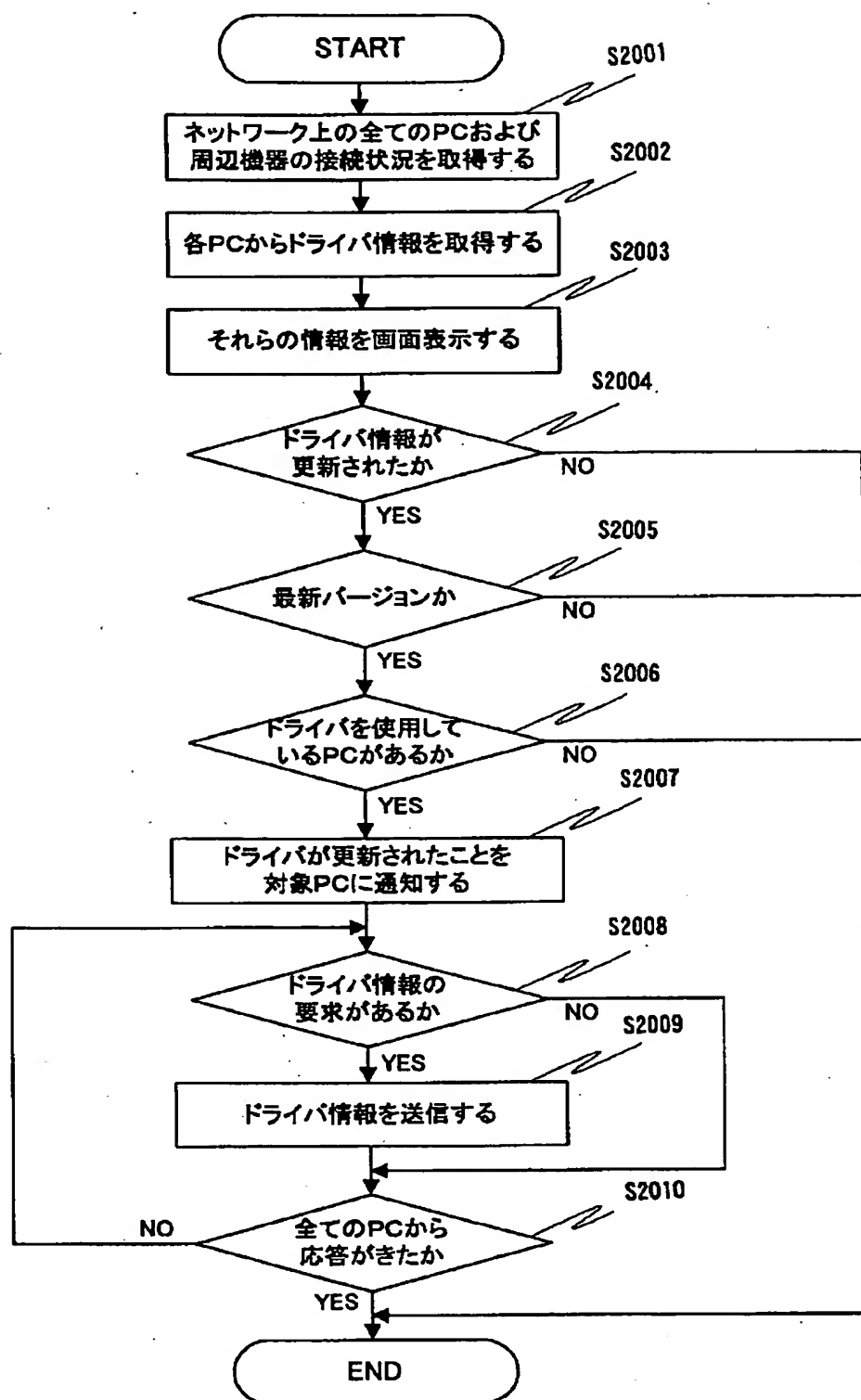
ドメイン名 (D): PC名 (P):

ドメイン2	SNPC01 1.00.00
	SNPC02 1.00.00
	SNPC03 1.00.00
	SNPC04 1.00.00
	SNPC05 1.00.00
	SNPC06 1.00.00
	SNPC07 1.00.00

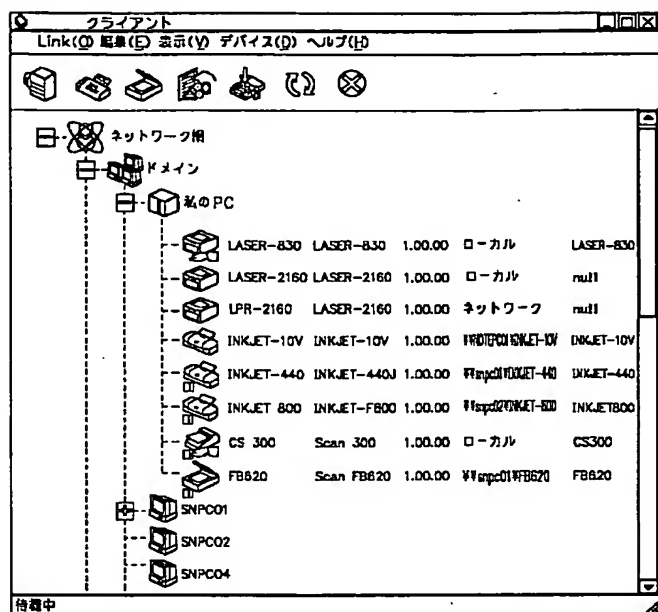
対象PC全てを選択

OK キャンセル

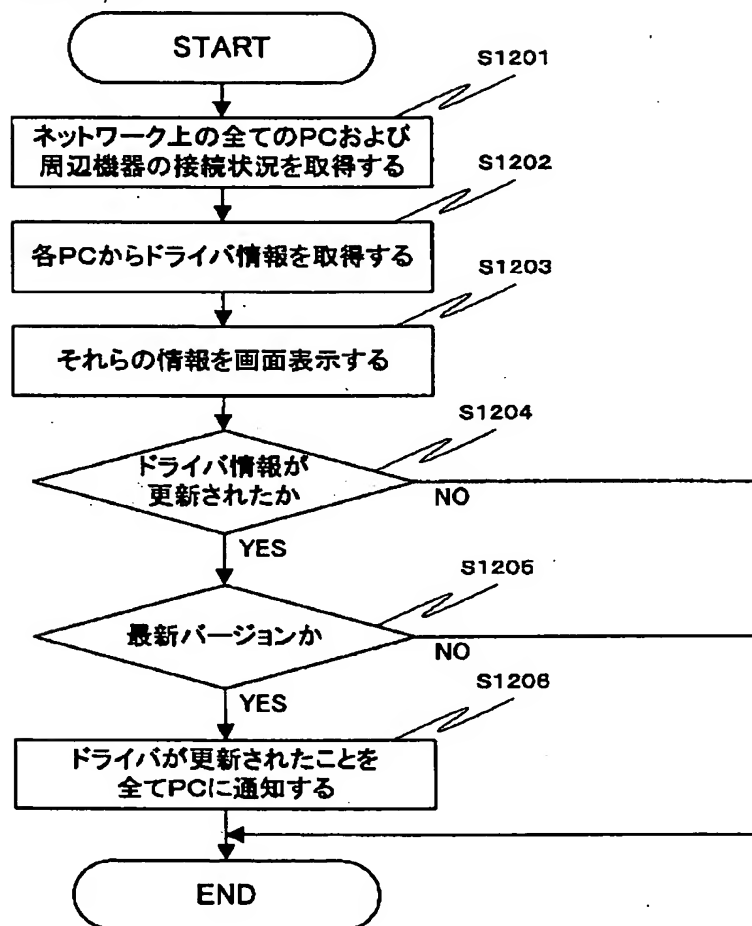
[Drawing 20]



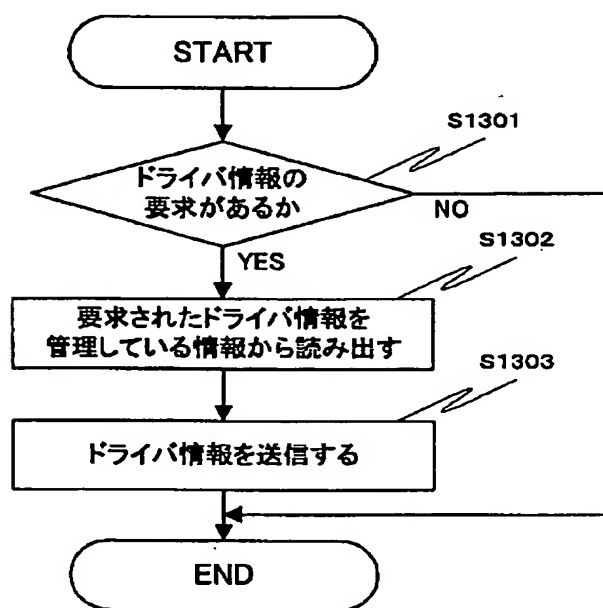
[Drawing 22]



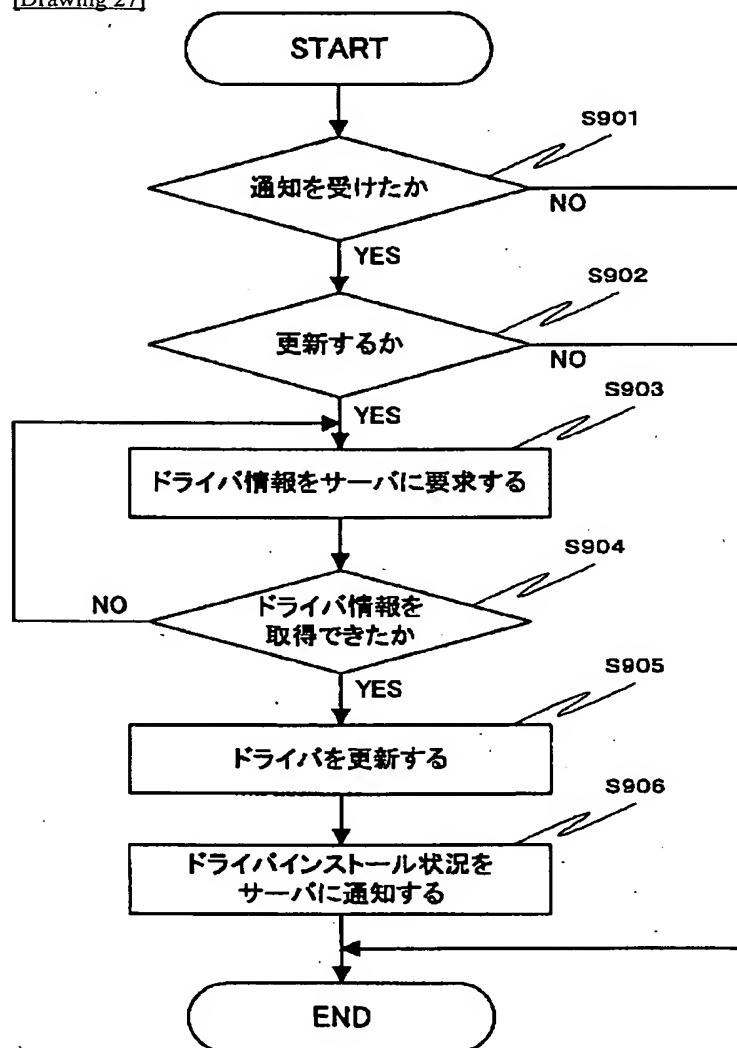
[Drawing 29]



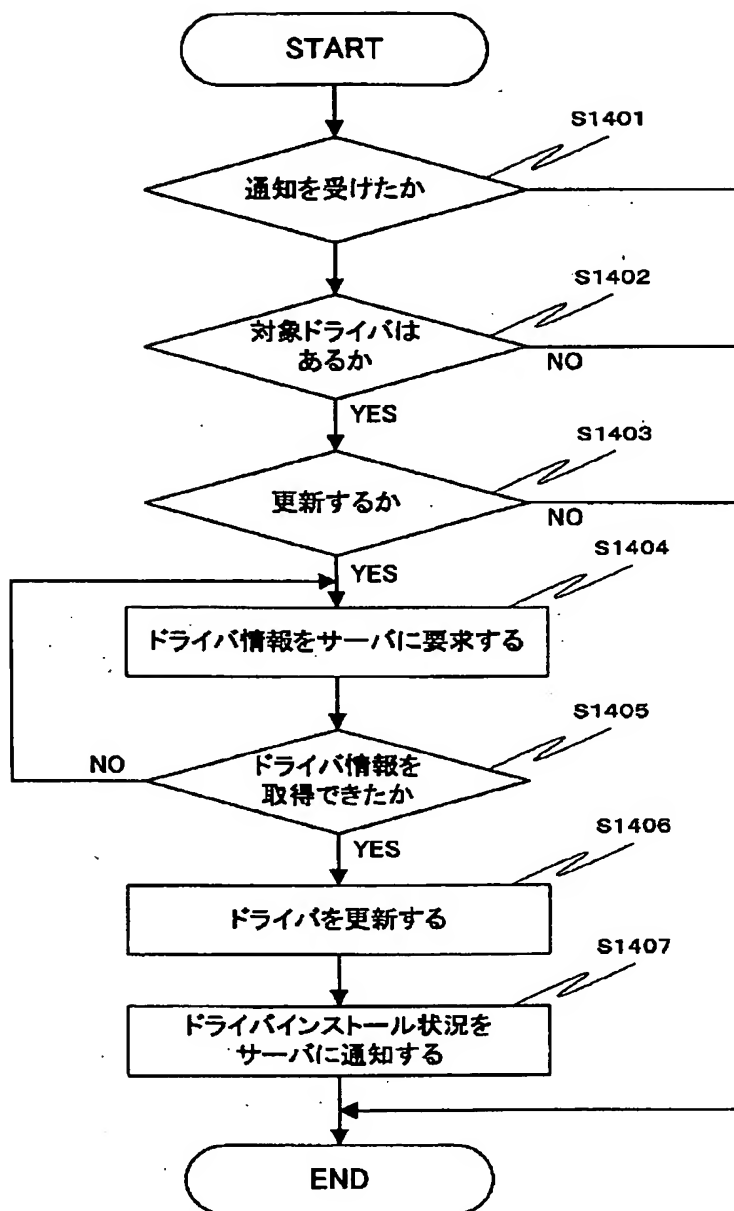
[Drawing 30]



[Drawing 27]



[Drawing 31]



[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CORRECTION OR AMENDMENT

[Kind of official gazette] Printing of amendment by the convention of 2 of Article 17 of Patent Law
 [Section partition] The 3rd partition of the 6th section
 [Publication date] March 14, Heisei 15 (2003. 3.14)

[Publication No.] JP,2002-251366,A (P2002-251366A)
 [Date of Publication] September 6, Heisei 14 (2002. 9.6)
 [Annual volume number] Open patent official report 14-2514
 [Application number] Application for patent 2001-355016 (P2001-355016)
 [The 7th edition of International Patent Classification]

G06F 13/10 320
 9/445

[FI]

G06F 13/10 320 A
 9/06 610 B
 610 D
 610 C
 610 L

[Procedure revision]

[Filing Date] December 10, Heisei 14 (2002. 12.10)

[Procedure amendment 1]

[Document to be Amended] Specification

[Item(s) to be Amended] Claim

[Method of Amendment] Modification

[Proposed Amendment]

[Claim(s)]

[Claim 1] It is the information processor which communicates with a peripheral device,

A setup information acquisition means to acquire the setup situation of the driver in said information processor,

The information processor characterized by having a judgment means to judge whether the peripheral device corresponding to the driver currently set up in said information processor exists.

[Claim 2] Said judgment means is an information processor according to claim 1 characterized by judging whether a peripheral device exists by whether it asks an output destination change port and there is any response.

[Claim 3] The information processor according to claim 1 characterized by having a driver deletion means to delete said driver when judged with the peripheral device corresponding to said driver not existing with said judgment means.

[Claim 4] According to one directions have a directions means to direct activation of deletion of the driver by said driver deletion means, and according [said driver deletion means] to said directions means, it is the information processor according to claim 3 characterized by performing deletion of two or more drivers.

[Claim 5] The information processor according to claim 3 characterized by having a check means to check to a user whether deletion is performed or not before deleting said driver with said driver deletion means.

[Claim 6] The information processor according to claim 3 characterized by having a deletion prohibition setting means to perform a prohibition setup of deletion of a desired driver.

[Claim 7] Said information processor communicates with an external information processor,

The information processor according to claim 1 which said judgment means judges that the peripheral device corresponding to said driver does not exist, and is characterized by having further a notice means of driver deletion to perform the notice of driver deletion which makes said external information processor delete a driver when said peripheral device is a local device.

[Claim 8] It is the information processing approach in the information processor which communicates with a peripheral device, The setup information acquisition procedure which acquires the setup situation of the driver in said information processor, The information processing approach characterized by having the judgment procedure of judging whether the peripheral device corresponding to the driver currently set up in said information processor existing.

[Claim 9] The information processing approach according to claim 8 characterized by judging whether a peripheral device exists by whether it asks an output destination change port and there is any response in said judgment procedure.

[Claim 10] The information processing approach according to claim 8 characterized by having the driver deletion procedure of deleting said driver when judged with the peripheral device corresponding to said driver not existing with said judgment procedure.

[Claim 11] The information processing approach according to claim 10 which performs the directions procedure of directing activation of deletion of a driver, in said driver deletion procedure, and is characterized by performing deletion of two or more drivers in said driver deletion procedure according to one directions by said directions procedure.

[Claim 12] The information processing approach according to claim 10 characterized by having the verification procedure which checks to a user whether deletion is performed or not before deleting said driver in said driver deletion procedure.

[Claim 13] The information processing approach according to claim 10 characterized by having the deletion prohibition configuration procedure which performs a prohibition setup of deletion of a desired driver.

[Claim 14] Said information processor communicates with an external information processor,

The information processing approach according to claim 8 which is judged that the peripheral device corresponding to said driver does not exist in said judgment procedure, and is characterized by having further the notice procedure of driver deletion of performing the notice of driver deletion which makes said external information processor deleting a driver when said peripheral device is a local device.

[Claim 15] It is the control program performed in the information processor which communicates with a peripheral device,

Setup information acquisition processing which acquires the setup situation of the driver in said information processor,

The control program characterized by performing judgment processing which judges whether the peripheral device corresponding to the driver currently set up in said information processor exists in said information processor.

[Claim 16] It is the information processor which communicates with an external information processor in a peripheral-device list,

A judgment means to judge whether the peripheral device corresponding to the driver currently set up exists,

The information processor characterized by equipping said external information processor with a notice means of driver deletion to perform the notice of driver deletion which makes the driver corresponding to said peripheral device delete when judged with the peripheral device corresponding to said driver not existing with said judgment means.

[Claim 17] It has further a setup information acquisition means to acquire the setup situation of the driver in said external information processor,

Said notice means of driver deletion is an information processor according to claim 16 characterized by performing the notice of deletion of a driver based on said setup information.

[Claim 18] The information processor according to claim 16 characterized by having a directions means to direct activation of the notice of deletion of the driver by said notice means of driver deletion.

[Claim 19] Said notice means of driver deletion is an information processor according to claim 18 characterized by performing the notice of driver deletion which makes said external information processor delete two or more drivers according to one directions by said directions means.

[Claim 20] The information processor according to claim 16 characterized by having a check means to check to a user whether deletion is performed or not before said notice means of driver deletion notified deletion of said driver.

[Claim 21] The information processor according to claim 16 characterized by having a deletion prohibition setting means to perform a prohibition setup of deletion of a desired driver.

[Claim 22] The information processor according to claim 16 which said judgment means judges with the peripheral device corresponding to said driver not existing, and is characterized by performing the notice of driver deletion which makes a driver delete to said external information processor when said peripheral device is a local device.

[Claim 23] It is the information processing approach in the information processor which communicates with an external information processor in a peripheral-device list,

The judgment procedure of judging whether the peripheral device corresponding to the driver currently set up existing,

The information processing approach characterized by having the notice procedure of driver deletion of performing the notice of driver deletion which makes said external information processor deleting the driver corresponding to said peripheral device when judged with the peripheral device corresponding to said driver not existing with said judgment procedure.

[Claim 24] It has further the setup information acquisition procedure which acquires the setup situation of the driver in said external information processor,

Said notice procedure of driver deletion is the information processing approach according to claim 23 characterized by performing the notice of deletion of a driver based on said setup information.

[Claim 25] The information processing approach according to claim 23 characterized by having the directions procedure of directing the notice of deletion of the driver by said notice procedure of driver deletion.

[Claim 26] Said notice procedure of driver deletion is the information processing approach according to claim 25 characterized by performing the notice of driver deletion which makes said external information processor perform deletion of two or more drivers according to one directions by said directions procedure.

[Claim 27] The information processing approach according to claim 23 characterized by having the verification procedure which checks to a user whether said external information processor is made to perform deletion before said notice procedure of driver deletion performs said notice of driver deletion.

[Claim 28] The information processing approach according to claim 23 characterized by having the deletion prohibition configuration procedure which performs a prohibition setup of deletion of a desired driver.

[Claim 29] The information processing approach according to claim 23 which judges with the peripheral device corresponding to said driver not existing in said judgment procedure, and is characterized by performing the notice of driver deletion which makes a driver delete to said external information processor when said peripheral device is a local device.

[Claim 30] It is the control program performed by the peripheral-device list in an external information processor and the information processor which communicates,

Judgment processing which judges whether the peripheral device corresponding to the driver currently set up exists,

The control program characterized by making said information processor perform notice processing of driver deletion in which the notice of driver deletion which makes said external information processor delete the driver corresponding to said peripheral device is performed when judged with the peripheral device corresponding to said driver not existing by said judgment processing.

[Claim 31] In the information processor which communicates with two or more external information processors in a peripheral-device list,

A recognition means to recognize the connection situation of the local device connected to said information processor,

The information processor characterized by having a notice means to perform the notice of deletion of the driver corresponding to said local device to an external information processor when said local device cannot be used from an external information processor according to the connection situation of the local device recognized by said recognition means.

[Claim 32] Said notice means is an information processor according to claim 31 characterized by performing the notices of deletion of a driver all at once to two or more external information processors.

[Claim 33] Said recognition means is an information processor according to claim 31 characterized by having the function to recognize whether said local device exists, and the function to recognize whether sharing setup of said local device is carried out.

[Claim 34] Said notice means is an information processor according to claim 33 characterized by performing the notice of deletion of said driver when recognized as said local device not existing with said recognition means, or when it has been recognized as sharing setup of said local device not being carried out.

[Claim 35] The information processor according to claim 31 characterized by recognizing said two or more information processors which are carrying out sharing setup of said local device based on the setup information which received from said two or more information processors.

[Claim 36] In the information processing approach in the information processor which communicates with two or more external information processors in a peripheral-device list,

The recognition procedure of recognizing the connection situation of the local device connected to said information processor, The information processing approach characterized by having the notice procedure of performing the notice of deletion of the driver corresponding to said local device to an external information processor when said local device cannot be used from an external information processor according to the connection situation of the local device recognized by said recognition procedure.

[Claim 37] Said notice procedure is the information processing approach according to claim 36 characterized by performing the notices of deletion of a driver all at once to two or more external information processors.

[Claim 38] Said recognition procedure is the information processing approach according to claim 36 characterized by having the procedure of recognizing whether said local device existing, and the procedure of recognizing whether sharing setup of said local device being carried out.

[Claim 39] Said notice procedure is the information processing approach according to claim 38 characterized by performing the notice of deletion of said driver when recognized as said local device not existing with said recognition procedure, or when it has been recognized as sharing setup of said local device not being carried out.

[Claim 40] The information processing approach according to claim 36 characterized by recognizing said two or more information processors which are carrying out sharing setup of said local device based on the setup information which received from said two or more information processors.

[Claim 41] In the control program in the information processor which communicates with two or more external information processors in a peripheral-device list,

Recognition processing which recognizes the connection situation of the local device connected to said information processor, The control program characterized by making said information processor perform notice processing which performs the notice of deletion of the driver corresponding to said local device to an external information processor when said local device cannot be used from an external information processor according to the connection situation of the local device recognized by said recognition processing.

[Claim 42] An input means to input the notice of deletion of the driver which includes assignment of a driver through an external network,

The information processor characterized by having a driver deletion means to delete the specified driver when said notice of deletion is inputted.

[Claim 43] The information processor according to claim 42 characterized by having a check means to check to a user whether deletion of said driver is performed before deleting said driver with said driver deletion means.

[Claim 44] The information processor according to claim 42 characterized by having a deletion prohibition setting means to set up prohibition of deletion of a desired driver.

[Claim 45] The information processor according to claim 44 characterized by having the control means controlled so that said driver deletion means does not delete a driver in said deletion prohibition setting means when prohibition of driver deletion is set up.

[Claim 46] The information processor according to claim 42 characterized by having communicated with the external information processor and having a collection means to collect setup information.

[Claim 47] The information processor according to claim 45 characterized by answering a demand from an external information processor and returning setup information.

[Claim 48] It has further a decision means to judge whether the driver specified by said notice of deletion is set up in the information processor,

Said driver deletion means is an information processor according to claim 42 characterized by disregarding said notice of deletion when the driver specified by said decision means is not set up.

[Claim 49] The input procedure which inputs the notice of deletion of the driver which includes assignment of a driver through an external network,

The information processing approach characterized by having the driver deletion procedure of deleting the specified driver when said notice of deletion is inputted.

[Claim 50] The information processing approach according to claim 49 characterized by having the verification procedure which checks to a user whether deletion of said driver is performed before deleting said driver with said driver deletion procedure.

[Claim 51] The information processing approach according to claim 49 characterized by having the deletion prohibition configuration procedure which sets up prohibition of deletion of a desired driver.

[Claim 52] The information processing approach according to claim 51 characterized by having the control procedure controlled by said driver deletion procedure in said deletion prohibition configuration procedure not to delete a driver when prohibition of driver deletion is set up.

[Claim 53] The information processing approach according to claim 49 characterized by communicating with an external information processor and having the collection procedure of collecting setup information.

[Claim 54] The information processing approach according to claim 53 characterized by answering a demand from an external information processor and returning setup information.

[Claim 55] It has further the decision procedure of judging whether the driver specified by said notice of deletion being set up in the information processor,

Said driver deletion procedure is the information processing approach according to claim 49 characterized by disregarding said notice of deletion when the driver specified by said decision procedure is not set up.

[Claim 56] Input process which inputs the notice of deletion of the driver which includes assignment of a driver through an external network,

The control program characterized by making an information processor perform driver deletion which deletes the specified driver when said notice of deletion is inputted.

[Claim 57] It is the network system which contains the second information processor in a peripheral-device list at the first information-processor list,

A judgment means to judge whether the peripheral device corresponding to the driver currently set up in said first information processor exists,

A notice means of driver deletion to perform the notice of driver deletion which makes said second information processor delete the driver corresponding to said peripheral device when judged with the peripheral device corresponding to said driver not existing with said judgment means,

An input means to input the notice of deletion of said driver into said second information processor through a network,

The network system characterized by having a driver deletion means to delete the specified driver when said notice of deletion is inputted.

[Translation done.]

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開 2002-251366

(P 2002-251366A)

(43) 公開日 平成14年9月6日 (2002.9.6)

(51) Int. Cl.⁷ 識別記号G 0 6 F 13/10
9/445

3 2 0

F I

G 0 6 F 13/10 3 2 0 A 5B014
9/06 6 1 0 B 5B076
6 1 0 D
6 1 0 C
6 1 0 L

テーマコード* (参考)

(全 2 3 頁)

審査請求 未請求 請求項の数 8 6 O L

(21) 出願番号 特願2001-355016 (P2001-355016)

(22) 出願日 平成13年11月20日 (2001. 11. 20)

(31) 優先権主張番号 特願2000-391201 (P2000-391201)

(32) 優先日 平成12年12月22日 (2000. 12. 22)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 町田 晴生

東京都大田区下丸子3丁目30番2号 キヤノ
ン株式会社内

(74) 代理人 100090273

弁理士 國分 孝悦

F ターム (参考) 5B014 EB03 FA11 HC07

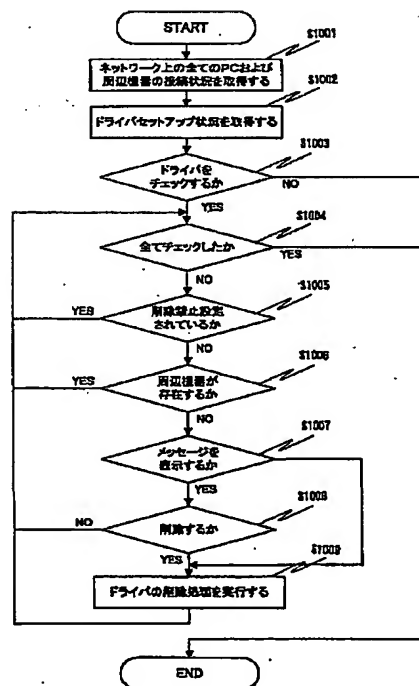
5B076 AA02 AA05 AA20 AC03

(54) 【発明の名称】 ネットワークシステム、情報処理装置、方法、及び制御プログラム

(57) 【要約】

【課題】 ネットワーク上で共有されている周辺機器が削除されたり、共有設定が解除されたりして使用できなくなった場合に、不要となったドライバを簡単な操作で削除できるようにする。

【解決手段】 ネットワーク上の全てのPC及び周辺機器の接続状況情報を取得するとともに、自機にインストールされている周辺機器のドライバのセットアップ状況情報を取得する。そして、ドライバのチェック指示があった場合、チェックするドライバが削除禁止設定されているか否かを判定し、禁止されていない場合は該ドライバに対応するデバイスが存在するか否かを判定して、対象デバイスが存在しなければ、ユーザに対して削除を実行するかどうかの指定を促すメッセージを表示した上で該ドライバの削除処理を実行する。



【特許請求の範囲】

【請求項 1】 周辺機器と通信する情報処理装置であって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、
前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段とを備えたことを特徴とする情報処理装置。

【請求項 2】 前記判定手段は、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在するか否かを判定することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 前記周辺機器並びに前記情報処理装置は、ネットワークを介して接続されていることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 4】 前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手段を備えたことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 5】 前記ドライバ削除手段によるドライバの削除の実行を指示する指示手段を備えたことを特徴とする請求項 4 に記載の情報処理装置。

【請求項 6】 前記ドライバ削除手段は、前記指示手段による一回の指示に応じて、複数のドライバの削除を実行することを特徴とする請求項 5 に記載の情報処理装置。

【請求項 7】 前記ドライバ削除手段により前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項 4 に記載の情報処理装置。

【請求項 8】 所望のドライバの削除の禁止設定を行う削除禁止設定手段を備えたことを特徴とする請求項 4 に記載の情報処理装置。

【請求項 9】 前記情報処理装置は外部情報処理装置と通信するものであり、

前記判定手段が、前記ドライバに対応する周辺機器が存在しないと判断し、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドライバ削除通知を行うドライバ削除通知手段をさらに備えたことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 10】 周辺機器と通信する情報処理装置における情報処理方法であって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、
前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段とを有することを特徴とする情報処理方法。

【請求項 11】 前記判定手段では、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在する

か否かを判定することを特徴とする請求項 10 に記載の情報処理方法。

【請求項 12】 前記周辺機器並びに前記情報処理装置は、ネットワークを介して接続されていることを特徴とする請求項 10 に記載の情報処理方法。

【請求項 13】 前記判定手順により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手段を有することを特徴とする請求項 10 に記載の情報処理方法。

10 【請求項 14】 前記ドライバ削除手順において、ドライバの削除の実行を指示する指示手順を行うことを特徴とする請求項 13 に記載の情報処理方法。

【請求項 15】 前記ドライバ削除手順では、前記指示手順による一回の指示に応じて、複数のドライバの削除を実行することを特徴とする請求項 14 に記載の情報処理方法。

20 【請求項 16】 前記ドライバ削除手順で前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認手順を有することを特徴とする請求項 13 に記載の情報処理方法。

【請求項 17】 所望のドライバの削除の禁止設定を行う削除禁止設定手段を有することを特徴とする請求項 13 に記載の情報処理方法。

30 【請求項 18】 前記情報処理装置は外部情報処理装置と通信するものであり、
前記判定手段で前記ドライバに対応する周辺機器が存在しないと判断され、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドライバ削除通知を行うドライバ削除通知手段をさらに有することを特徴とする請求項 10 に記載の情報処理方法。

【請求項 19】 周辺機器と通信する情報処理装置において実行される制御プログラムであって、
前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、
前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段とを前記情報処理装置において実行させることを特徴とする制御プログラム。

40 【請求項 20】 前記判定処理では、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在するか否かを判定するよう前記情報処理装置を実行させることを特徴とする請求項 19 に記載の制御プログラム。

【請求項 21】 前記周辺機器並びに前記情報処理装置は、ネットワークを介して接続されていることを特徴とする請求項 19 に記載の制御プログラム。

50 【請求項 22】 前記判定処理により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除処理をさらに前記情報処理装置に実行させることを特徴とする請求項 19 に記載の制御プログラム。

載の制御プログラム。

【請求項 2 3】 前記ドライバ削除処理において、ドライバの削除の実行を指示する指示処理をさらに前記情報処理装置に実行させることを特徴とする請求項 2 2 に記載の制御プログラム。

【請求項 2 4】 前記ドライバ削除処理では、前記指示処理による一回の指示に応じて、複数のドライバの削除を行うよう前記情報処理装置を実行させることを特徴とする請求項 2 3 に記載の制御プログラム。

【請求項 2 5】 前記ドライバ削除処理で前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認処理をさらに前記情報処理装置に実行させることを特徴とする請求項 2 2 に記載の制御プログラム。

【請求項 2 6】 所望のドライバの削除の禁止設定を行う削除禁止設定処理をさらに前記情報処理装置に実行させることを特徴とする請求項 2 2 に記載の制御プログラム。

【請求項 2 7】 前記情報処理装置は外部情報処理装置と通信するものであり、前記判定処理で前記ドライバに対応する周辺機器が存在しないと判断され、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドライバ削除通知を行うドライバ削除通知処理をさらに前記情報処理装置に実行させることを特徴とする請求項 1 9 に記載の制御プログラム。

【請求項 2 8】 周辺機器並びに外部情報処理装置と通信する情報処理装置であって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段とを備えたことを特徴とする情報処理装置。

【請求項 2 9】 前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段をさらに備え、前記ドライバ削除通知手段は、前記セットアップ情報に基づいて、ドライバの削除通知を行うことを特徴とする請求項 2 8 に記載の情報処理装置。

【請求項 3 0】 前記ドライバ削除通知手段によるドライバの削除通知の実行を指示する指示手段を備えたことを特徴とする請求項 2 8 に記載の情報処理装置。

【請求項 3 1】 前記ドライバ削除通知手段は、前記指示手段による一回の指示に応じて、前記外部情報処理装置に複数のドライバを削除させるドライバ削除通知を行うことを特徴とする請求項 3 0 に記載の情報処理装置。

【請求項 3 2】 前記ドライバ削除通知手段により前記ドライバの削除を通知する前に、削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする

請求項 2 8 に記載の情報処理装置。

【請求項 3 3】 所望のドライバの削除の禁止設定を行う削除禁止設定手段を備えたことを特徴とする請求項 2 8 に記載の情報処理装置。

【請求項 3 4】 前記判定手段が、前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うことを特徴とする請求項 2 8 に記載の情報処理装置。

10 【請求項 3 5】 周辺機器並びに外部情報処理装置と通信する情報処理装置における情報処理方法であって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手順と、前記判定手順により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手順とを有することを特徴とする情報処理方法。

20 【請求項 3 6】 前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手順をさらに有し、前記ドライバ削除通知手順は、前記セットアップ情報に基づいて、ドライバの削除通知を行うことを特徴とする請求項 3 5 に記載の情報処理方法。

【請求項 3 7】 前記ドライバ削除通知手順によるドライバの削除の通知を指示する指示手順を有することを特徴とする請求項 3 5 に記載の情報処理方法。

30 【請求項 3 8】 前記ドライバ削除通知手順は、前記指示手順による一回の指示に応じて、複数のドライバの削除を前記外部情報処理装置に実行させるドライバ削除通知を行うことを特徴とする請求項 3 7 に記載の情報処理方法。

【請求項 3 9】 前記ドライバ削除通知手順により前記ドライバ削除通知を行う前に、前記外部情報処理装置に削除を実行させるか否かをユーザに確認する確認手順を有することを特徴とする請求項 3 5 に記載の情報処理方法。

40 【請求項 4 0】 所望のドライバの削除の禁止設定を行う削除禁止設定手順を有することを特徴とする請求項 3 5 に記載の情報処理方法。

【請求項 4 1】 前記判定手順で前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うことを特徴とする請求項 3 5 に記載の情報処理方法。

50 【請求項 4 2】 周辺機器並びに外部情報処理装置と通信する情報処理装置において実行される制御プログラムであって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定処理と、

前記判定処理により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知処理とを前記情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 4 3】 前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得処理を前記情報処理装置に実行させ、

前記ドライバ削除通知処理は、前記セットアップ情報に基づいて、ドライバの削除通知を行うよう前記情報処理装置を実行させることを特徴とする請求項 4 2 に記載の制御プログラム。

【請求項 4 4】 前記ドライバ削除通知処理によるドライバの削除通知の実行を指示する指示処理をさらに前記情報処理装置に実行させることを特徴とする請求項 4 2 に記載の制御プログラム。

【請求項 4 5】 前記ドライバ削除通知処理は、前記指示処理による一回の指示に応じて、前記外部情報処理装置に複数のドライバの削除を行わせるドライバ削除通知を行うよう前記情報処理装置を実行させることを特徴とする請求項 4 4 に記載の制御プログラム。

【請求項 4 6】 前記ドライバ削除通知処理により前記ドライバ削除通知を行う前に、前記外部情報処理装置に削除を実行させるか否かをユーザに確認する確認処理をさらに情報処理装置に実行させることを特徴とする請求項 4 2 に記載の制御プログラム。

【請求項 4 7】 所望のドライバの削除の禁止設定を行う削除禁止設定処理をさらに前記情報処理装置に実行させることを特徴とする請求項 4 2 に記載の制御プログラム。

【請求項 4 8】 前記判定処理で、前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うよう前記情報処理装置を実行させることを特徴とする請求項 4 2 に記載の制御プログラム。

【請求項 4 9】 周辺装置並びに複数の外部情報処理装置と通信する情報処理装置において、前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手段と、

前記認識手段により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手段とを備えたことを特徴とする情報処理装置。

【請求項 5 0】 前記通知手段は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うことを特徴とする請求項 4 9 に記載の情報処理装置。

【請求項 5 1】 前記認識手段は、前記ローカルデバイスが存在するか否かを認識する機能、及び、前記ローカ

ルデバイスが共有設定されているか否かを認識する機能を備えたことを特徴とする請求項 4 9 に記載の情報処理装置。

【請求項 5 2】 前記通知手段は、前記認識手段により前記ローカルデバイスが存在しないと認識された場合、又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うことを特徴とする請求項 5 1 に記載の情報処理装置。

10 【請求項 5 3】 前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識することを特徴とする請求項 4 9 に記載の情報処理装置。

【請求項 5 4】 周辺装置並びに複数の外部情報処理装置と通信する情報処理装置における情報処理方法において、

前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手順と、

前記認識手順により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手順とを有することを特徴とする情報処理方法。

20 【請求項 5 5】 前記通知手順は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うことを特徴とする請求項 5 4 に記載の情報処理方法。

【請求項 5 6】 前記認識手順は、前記ローカルデバイスが存在するか否かを認識する手順、及び、前記ローカルデバイスが共有設定されているか否かを認識する手順を有することを特徴とする請求項 5 4 に記載の情報処理方法。

30 【請求項 5 7】 前記通知手順は、前記認識手段により前記ローカルデバイスが存在しないと認識された場合、又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うことを特徴とする請求項 5 6 に記載の情報処理方法。

【請求項 5 8】 前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識することを特徴とする請求項 5 4 に記載の情報処理方法。

40 【請求項 5 9】 周辺装置並びに複数の外部情報処理装置と通信する情報処理装置における制御プログラムにおいて、

前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識処理と、

前記認識処理により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知処理とを前記情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 60】 前記通知処理は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うよう前記情報処理装置を実行させることを特徴とする請求項 59 に記載の制御プログラム。

【請求項 61】 前記認識処理は、前記ローカルデバイスが存在するか否かを認識する処理、及び、前記ローカルデバイスが共有設定されているか否かを認識する処理をさらに前記情報処理装置に実行させることを特徴とする請求項 59 に記載の制御プログラム。

【請求項 62】 前記通知処理は、前記認識処理により前記ローカルデバイスが存在しないと認識された場合、又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うよう前記情報処理装置を実行させることを特徴とする請求項 61 に記載の制御プログラム。

【請求項 63】 前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識するよう前記情報処理装置を実行させることを特徴とする請求項 59 に記載の制御プログラム。

【請求項 64】 外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手段と、前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えたことを特徴とする情報処理装置。

【請求項 65】 前記ドライバ削除手段により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項 64 に記載の情報処理装置。

【請求項 66】 所望のドライバの削除の禁止を設定する削除禁止設定手段を備えたことを特徴とする請求項 64 に記載の情報処理装置。

【請求項 67】 前記削除禁止設定手段において、ドライバ削除の禁止が設定されている場合に、前記ドライバ削除手段がドライバを削除しないように制御する制御手段を備えたことを特徴とする請求項 66 に記載の情報処理装置。

【請求項 68】 前記情報処理装置は外部情報処理装置と通信して、セットアップ情報の収集を行う収集手段を備えたことを特徴とする請求項 64 に記載の情報処理装置。

【請求項 69】 前記情報処理装置は、外部情報処理装置からの要求に応答して、セットアップ情報を返送することを特徴とする請求項 67 に記載の情報処理装置。

【請求項 70】 前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手段をさらに備え、前記ドライバ削除手段は、前記判断手段により指定されたドライバがセットアップされていない場合には、前記削除通知を無視することを特徴とする請求項 64 に記載

の情報処理装置。

【請求項 71】 外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手順と、前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手順とを有することを特徴とする情報処理方法。

【請求項 72】 前記ドライバ削除手順により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認手順を有することを特徴とする請求項 71 に記載の情報処理方法。

【請求項 73】 所望のドライバの削除の禁止を設定する削除禁止設定手順を有することを特徴とする請求項 71 に記載の情報処理方法。

【請求項 74】 前記削除禁止設定手順において、ドライバ削除の禁止が設定されている場合に、前記ドライバ削除手順ではドライバを削除しないように制御する制御手順を有することを特徴とする請求項 73 に記載の情報処理方法。

【請求項 75】 前記情報処理装置は外部情報処理装置と通信して、セットアップ情報の収集を行う収集手順を有することを特徴とする請求項 71 に記載の情報処理方法。

【請求項 76】 前記情報処理装置は、外部情報処理装置からの要求に応答して、セットアップ情報を返送することを特徴とする請求項 75 に記載の情報処理方法。

【請求項 77】 前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手順をさらに有し、

前記ドライバ削除手順は、前記判断手順により指定されたドライバがセットアップされていない場合には、前記削除通知を無視することを特徴とする請求項 71 に記載の情報処理方法。

【請求項 78】 外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力処理と、前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除処理とを情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 79】 前記ドライバ削除処理により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認処理を前記情報処理装置に実行させることを特徴とする請求項 78 に記載の制御プログラム。

【請求項 80】 所望のドライバの削除の禁止を設定する削除禁止設定処理を前記情報処理装置に実行させることを特徴とする請求項 78 に記載の制御プログラム。

【請求項 81】 前記削除禁止設定処理において、ドライバ削除の禁止が設定されている場合に、前記ドライバ削除処理ではドライバを削除しないように制御する制御処理を前記情報処理装置に実行させることを特徴とする請求項 78 に記載の制御プログラム。

【請求項 8 2】 前記情報処理装置は外部情報処理装置と通信して、セットアップ情報の収集を行う収集処理を前記情報処理装置に実行させることを特徴とする請求項 7 8 に記載の制御プログラム。

【請求項 8 3】 前記情報処理装置は、外部情報処理装置からの要求に回答して、セットアップ情報を返送するよう前記情報処理装置を実行させることを特徴とする請求項 8 2 に記載の制御プログラム。

【請求項 8 4】 前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手順をさらに前記情報処理装置に実行させ、前記ドライバ削除処理は、前記判断処理により指定されたドライバがセットアップされていない場合には、前記削除通知を無視するよう前記情報処理装置を実行させることを特徴とする請求項 7 8 に記載の制御プログラム。

【請求項 8 5】 周辺機器並びに第一の情報処理装置並びに第二の情報処理装置を含むネットワークシステムであって、

前記第一の情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、

前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記第二の情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段と、

ネットワークを介して前記ドライバの削除通知を前記第二の情報処理装置に入力する入力手段と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えたことを特徴とするネットワークシステム。

【請求項 8 6】 請求項 1 9 ~ 2 7、4 2 ~ 4 8、5 9 ~ 6 3、7 8 ~ 8 4 のいずれかが 1 項に記載の制御プログラムを格納したことを特徴とするコンピュータ読み取り可能な記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ネットワークシステム、情報処理装置、方法、及び制御プログラムに係り、ネットワーク上で共有された情報処理装置や周辺機器の表示や管理等を行うものに関するものである。

【0002】

【従来の技術】近年、パーソナルコンピュータ（以下、PC と称する）や、プリンタ、スキャナ、デジタルカメラ等の周辺機器の普及と共に、LAN 等のネットワークも普及してきており、ネットワーク上で複数の PC によりプリンタ、モデム、スキャナ等を共有するニーズも増えてきた。また、これらの周辺機器を使用するためのドライバも数多く提供され、そのインストール方法も提供されてきた。

【0003】

【発明が解決しようとする課題】しかし、周辺機器を性能上の問題で買い替えたり、使用する上で接続する PC を変更したりした場合に、これらの周辺機器のドライバが削除されないままとなることが多かった。そのため、実際に周辺機器を使用する際に、不必要なドライバまでリスト表示されるために選択しにくくなったり、どのドライバが有効であるか分からなくなったりすることがあり、作業効率が悪くなってしまうという問題があった。

【0004】本発明は上記のような点に鑑みてなされたものであり、ネットワーク上で共有されている周辺機器が削除されたり、共有設定が解除されたりして使用できなくなった場合に、不要となったドライバを簡単な操作で削除できるようにすることを目的とする。

【0005】

【課題を解決するための手段】上記課題を解決するための手段として、本発明の情報処理装置について述べると、本発明の情報処理装置は、周辺機器と通信する情報処理装置であって、前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段とを備えた点に特徴を有する。

【0006】また、本発明の他の情報処理装置は、周辺機器並びに外部情報処理装置と通信する情報処理装置であって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段とを備えた点に特徴を有する。

【0007】また、本発明の他の情報処理装置は、周辺装置並びに複数の外部情報処理装置と通信する情報処理装置において、前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手段と、前記認識手段により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手段とを備えた点に特徴を有する。

【0008】また、本発明の他の情報処理装置は、外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手段と、前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えた点に特徴を有する。

【0009】

【発明の実施の形態】以下、図面を参照して、本発明のネットワークシステム、情報処理装置、方法、及び制御プログラムの実施の形態について説明する。

【0010】図 1 は、本実施の形態の情報処理装置の構

10

20

30

40

50

成を説明するブロック図である。同図において、1はシステムバスであり、当該システムバス1を介して以下に述べる各構成ブロックが接続されている。

【0011】2はCPU (Central Processing Unit) である。3はプログラムメモリ (以下、P MEMと称する) であり、後述する処理のためのプログラムを適宜ハードディスク10から選択/読み込みし、CPU 2にて実行する。また、キーボード12から入力されたデータは、テキストメモリでもあるP MEMにコード情報として格納される。

【0012】4は通信制御部であり、通信ポート5における入出力データの制御を行う。通信ポート5から出力された信号は、通信回線6を経由して、ネットワーク上の他の装置 (図1では符号7を付す) の通信ポートに伝えられる。ネットワーク上で共有されているプリンタやスキャナ等とのやり取りは、この通信制御部4を介して行われる。本実施の形態では、LAN等のネットワークに関して記述するが、この通信制御部4に接続される通信ポート5及び通信回線6が一般の公衆回線であっても本発明が適応されることは言うまでもない。

【0013】8は外部記憶装置制御部である。9、10はデータファイル用のディスクであり、例えば9はフレキシブルディスクFDであり、10はハードディスクHDである。

【0014】11は入力制御部であり、キーボード12、マウス13等の入力装置が接続される。ユーザは、キーボード12を操作することによりシステムの動作指令等を行うことができる。また、マウス13はCRT16上で画像情報を加工指示するためのポインティング・デバイスPDとして機能し、これによりCRT16上のカーソルをX、Y方向任意に移動してコマンドメニュー上のコマンド・アイコンを選択して処理の指示を行う他、編集対象の指示、描画位置の指示等も行うことができる。

【0015】14はビデオイメージメモリ (VRAM) であり、15は表示出力制御部であり、16はCRT (Cathode-Ray Tube) である。CRT16に表示されるデータは、VRAM14上にビットマップデータとして展開される。

【0016】17はプリンタ制御部であり、接続されているプリンタ18に対するデータの出力制御を行う。1Aは、画像読み取り装置制御部であり、接続されている画像読み取り機器1Bの制御を行う。

【0017】なお、本実施の形態でP MEM3に記憶しているプログラムは、ハードディスク10やフレキシブルディスク9等の記憶媒体に記憶されていてもよい。さらに、ネットワークに接続されている他の装置上に記憶されていてもよい。

【0018】以下、図2～9を参照して、周辺機器のドライバのセットアップを行う場合について説明する。図

2は、ドライバのセットアップを行う処理動作を示すフローチャートである。まず、ステップS201で、ネットワーク上の全てのPC及び周辺機器の接続状況情報を取得し、その情報を記憶する。

【0019】次に、ステップS202で、各PCの周辺機器のドライバのセットアップ状況情報を取得する。周辺機器のドライバとは、プリンタ、スキャナ、デジタルカメラ、FAX等のドライバのことである。セットアップ状況情報の取得方法として、例えば、各PC上で自機にインストールされているドライバ情報を取得するモジュールを起動させることで情報を収集して、それらの情報をネットワークを通じて通信することが考えられる。

【0020】図3は、セットアップされているドライバ情報構造体の一例を示す図である。PC名称、IPアドレス、OS種類、ユーザ名、デバイスドライバ数、及び各デバイスドライバの情報から構成される。各デバイスドライバ情報は、プリンタやスキャナといったデバイス種類、ドライバ名、バージョン情報、出力ポート、共有情報で構成される。例えば、1番目に登録されているプリンタは、ドライバ名がprinter 2000、バージョンが1.00.00、出力ポートがローカル、共有情報が共有設定 (ON) となっている。

【0021】図4は、ネットワーク上のPC及び周辺機器を表示する画面の一例を示す図である。401はメニュー、402はツールバー、403はネットワーク上のPC及び周辺機器を表すアイコンが表示されるメインウィンドウである。

【0022】402a～402gは、ツールバー402においてPC及び周辺機器を操作して行う各機能を表すアイコンである。例えば、402aは選択されたスキャナから画像データを読み込み、該画像データを選択されたプリンタに出力するコピー機能を実行するためのアイコンである。402bはFAX機能、402cは画像データ読み込み機能、402dは画像データを読み込み、OCR処理を行う機能、402eはFAX受信データ及び配信データを管理する機能、402fは情報の更新処理を行う機能、402gは更新処理を中止する機能を実行するためのアイコンである。

【0023】403a～403aeはネットワーク上で共有されているPC及び周辺機器を表すアイコンである。これらのアイコン403a～403aeは、PC、プリンタ、スキャナ、FAXモデムといったデバイス種類、或いは「処理中」、「エラー発生」等のステータスによってアイコンが変更されて表示される。403aは自機を表すアイコン、403bは自機がログオンしているドメインを表すアイコンである。自機は特別な機器であるために、他のPCとは区別して表示される。

【0024】また、アイコン403adのようにネットワーク上では共有されているが、ドライバがインストールされていないPC及び周辺機器はアイコンがグレー表

示される。また、アイコン403d、403eのように接続機器があるが展開して表示されていない場合には「+」マークが、アイコン403abのように展開されている場合には「-」マークが表示される。アイコン403c、403gのように接続機器が何もない場合にはマークが付加されない。

【0025】このように該画面でネットワーク上の全てのPC及び周辺機器の接続状態、さらにはステータスを確認することができる。この例では、画面の都合上、全てのアイコンが表示されていないが、画面横に配置されているスクロールバーを使って全てのPC及び周辺機器を確認することができる。

【0026】図5は、ネットワーク上の各PCのドライバセットアップ状況を表示する画面の一例を示す図である。501はメニュー、502はツールバー、503はネットワーク上のPC及び周辺機器を表すアイコンが表示されるメインウィンドウである。

【0027】502a~502gは、ツールバー502においてPC及び周辺機器を操作して行う各機能を表すアイコンであり、その内容は、図4に示すアイコン402a~402gと同様である。503a~503nはネットワーク上で共有されているPC及び周辺機器を表すアイコンである。ここで、周辺機器のドライバ情報が表示され、確認することができる。

【0028】図2のフローチャートに説明を戻すと、ステップS203で、周辺機器のドライバをセットアップするか否かを判定する。例えば、図4においてプリンタ403adを選択して、メニューからドライバのセットアップ指示を行うことができる。ここで、ドライバのセットアップ指示がなかった場合は、この処理を終了する。

【0029】ドライバのセットアップ指示があった場合、ステップS204に移行し、セットアップするドライバを指示する。図6はセットアップするドライバを選択するための画面の一例であり、ここで製造元を選択して、対象プリンタを指示する。また、ユーザがセットアップ情報のあるフォルダを指示することもできる。

【0030】さらに、ステップS205で、ドライバをセットアップするPCを選択する。図7はドライバをセットアップするPCを選択するための画面の一例であり、ここでドメイン、及びPCを選択する。この際、複数のPCを選択することができる。

【0031】そして、OKの指示があると、ステップS206に移行し、指示されたドライバを指示されたPCにセットアップする必要があるかを、それぞれのPCからのドライバ情報から判定する。指示されたドライバが既にインストールされている等してセットアップする必要がない場合は、インストール処理を行わない。

【0032】セットアップする必要がある場合、ステップS207に移行し、ステップS205で指示されたP

Cに対してステップS204で指示されたドライバのセットアップ処理を実行するように指示が送られる。そして、その指示に従ってドライバのセットアップ処理が行われる。図8はドライバセットアップ指示構造体の一例であり、デバイス種類、ドライバ名、バージョン情報、出力ポート、セットアップ情報等から構成される。なお、セットアップ情報はこの構造体と共に送られてもよいが、ネットワーク上の共有フォルダに記憶しておいてそのパス名を指示するようにしてもよい。

10 【0033】次に、ステップS208で、ドライバのセットアップ処理が正常終了したか否かを判定する。判定方法としては、ドライバのインストール処理が正常に行われたかどうか判定すると同時に、実際にそのドライバを使用して正常動作するかどうかで判定する。例えば、ドライバが正常にインストールできた場合には、セットアップ指示をしたPCに対して正常終了したメッセージが送信されるようにして、画面上で確認できるようにする。

20 【0034】プリンタの場合には、さらにテストプリントを実行することで、その印刷結果を見て、正常にインストールできたかどうかを判定できるようにしてもよい。図9は、ドライバのセットアップ確認印刷の一例を示す。このように、出力元PC名、ドライバ名、バージョン情報等を印刷することで、どのPC上でセットアップ処理が正常に完了したかを確認することができる。正常終了した場合は、処理を終了する。

30 【0035】正常終了しなかった場合、ステップS209に移行し、正常終了しなかったPCに対して再度実行指示をするかを判定する。再実行の指示があった場合は、ステップS204に戻り、再実行の指示がなかった場合、処理を終了する。

【0036】以上述べたように、ネットワーク上で共有されているPC及び周辺機器が表示されている画面において、周辺機器のドライバがインストールされていない場合、簡単な操作でドライバをインストールできると共に、実際にインストールされたことを確認することができ、ネットワーク上での作業効率を大幅に向上させることができる。

40 【0037】以下、図10~19を参照して、周辺機器のドライバのチェック、さらには削除を行う場合について説明する。図10は、ドライバのチェック、さらには削除を行う処理動作を示すフローチャートである。まず、ステップS1001で、ネットワーク上の全てのPC及び周辺機器の接続状況情報を取得し、その情報を記憶する。

50 【0038】次に、ステップS1002で、自機にインストールされている周辺機器のドライバのセットアップ状況情報を取得する。周辺機器のドライバとは、プリンタ、スキャナ、デジタルカメラ、FAX等のドライバのことである。

【0039】図11は、セットアップされているドライバ情報構造体の一例を示す図である。PC名称、IPアドレス、OS種類、ユーザ名、デバイスドライバ数、及び各デバイスドライバの情報から構成される。各デバイスドライバ情報は、プリンタやスキャナといったデバイス種類、ドライバ名、バージョン情報、出力ポート、共有情報で構成される。例えば、1番目に登録されているプリンタは、ドライバ名がLASER-830、バージョンが1.00.00、出力ポートがローカル接続、共有情報が共有設定(ON)となっている。

【0040】図12は、ネットワーク上のPC及び周辺機器を表示した画面の一例を示す図である。301はメニュー、302はツールバー、303はPC及び周辺機器を表すアイコンが表示されるメインウィンドウである。

【0041】302a~302iは、ツールバー302においてPC及び周辺機器を操作して行う各機能を表すアイコンである。例えば、302aは選択されたスキャナから画像データの読み込み、該画像データを選択されたプリンタに出力するコピー機能を実行するためのアイコンである。302bはFAX機能、302cは画像データ読み込み機能、302dは画像データを読み込み、OCR処理を行う機能、302eはFAX受信データ及び配信データを管理する機能、302fは表示切り替え機能、302gは表示画面編集機能、302hは情報の更新処理を行う機能、302iは更新処理を中止する機能を実行するためのアイコンである。

【0042】303a~303qはネットワーク上で共有されているPC及び周辺機器を表すアイコンである。これらのアイコン303a~303qは、PC、プリンタ、スキャナ、FAXモデムといったデバイス種類、或いは「処理中」、「エラー発生」等のステータスによってアイコンが変更されて表示される。303cは自機を表すアイコン、303bは自機がログオンしているドメインを表すアイコンである。自機は特別な機器であるために、他のPCとは区別して先頭に表示される。その他のPCについては、アルファベット順に昇順或いは降順で表示される。

【0043】また、アイコン303pのようにネットワーク上では共有されているが、ドライバがインストールされていないPC及び周辺機器はアイコンがグレー表示される。また、アイコン303jのように接続機器があるが展開して表示されていない場合には「+」マークが、アイコン303h、303k、303nのように展開されている場合には「-」マークが表示される。アイコン303qのように接続機器が何もない場合はマークが付加されない。

【0044】このように該画面でネットワーク上の全てのPC及び周辺機器の接続状態、さらにはステータスを確認することができる。この例では、画面の都合上、全

てのアイコンが表示されていないが、画面横に配置されているスクロールバーを使って全てのPC及び周辺機器を確認することができる。

【0045】図13は、図12に示す状態から、アイコン303hで表されるPCに接続されてネットワーク上で共有設定されていたアイコン303iで表されるプリンタの共有設定が解除された、或いはドライバが削除された場合に表示される画面の一例を示す図である。同図に示すように、アイコン303iには、「使用不可」を示す表示「×」印が表示される。この状態では、ドライバがインストールされているが、実際の出力先が存在しないので出力できない。

【0046】図10のフローチャートに説明を戻すと、ステップS1003で、周辺機器のドライバをチェックするか否かを判定する。例えば、図14に示すような画面が表示されて、チェックするデバイスを選択することでドライバのチェック指示を行うことができる。この画面で、チェックした際に対応するデバイスがないために不要だと判定された場合、メッセージ表示するか否かの設定も併せて指定することができる。ここで、ドライバのチェック指示がなかった場合は、この処理を終了する。

【0047】ドライバのチェック指示があった場合、ステップS1004に移行し、チェック対象とされたドライバ全てをチェックしたか否かを判定する。全てのドライバについてチェックした場合は、この処理を終了する。

【0048】全てのドライバをチェックしていない場合、ステップS1005に移行し、チェックするドライバが削除禁止設定されているか否かを判定する。例えば、通常良く使うプリンタ等でチェック処理を実行したときに、たまたま電源が入っていない等していたために対象デバイスがないと判定されて削除されるのでは、再度ドライバをインストールしなおさなければならず、ユーザにとって不便である。

【0049】そこで、削除したくないドライバに対して削除禁止設定を行えるようにする。図15はドライバの削除禁止設定を行うための画面の一例であり、ここで削除を禁止したいドライバを選択して、禁止リストに追加登録することで削除禁止設定を行うことができる。逆に、削除禁止を解除したいドライバを選択して、禁止リストから削除することもできる。

【0050】上記ステップS1005で、禁止されていた場合は、ステップS1004に戻り、次のドライバをチェックする。禁止されていない場合、ステップS1006に移行し、ドライバに対応するデバイスが存在するか否かを判定する。例えば、プリンタならば、出力ポート先に問い合わせたて応答があるかどうかで判定する。スキャナ等に対しては、入力ポートに対して同様の処理を行ってチェックする。対象デバイスが存在する場合は、

ステップS1004に戻り、次のドライバをチェックする。

【0051】対象デバイスが存在しない場合、ステップS1007に移行し、削除メッセージを表示するか否かを判定する。この判定は、ステップS1003で設定された情報に従って行われる。ここで、メッセージを表示するように設定されていた場合は、ステップS1008に移行する。メッセージを表示しないように設定されていた場合、ステップS1009に移行する。

【0052】ステップS1008では、ユーザに対して削除を実行するかどうかの指定を促すメッセージが表示される。図16は、削除するドライバがあった場合に表示されるメッセージの一例である。このように対象ドライバが表示されて、削除するかどうかの指示を行う。ここで、「はい」が指示された場合は、ステップS1009に移行する。「いいえ」が指示された場合、削除処理がキャンセルされ、ステップS1004に戻り、次のドライバをチェックする。

【0053】ステップS1009では、指定されたドライバの削除処理を実行する。そして処理が終わった後にステップS1004に戻り、次のドライバをチェックする。

【0054】図10に示したフローチャートでは、デバイスを共有設定されているコンピュータでドライバのチェックを行っているが、デバイスを共有設定しているコンピュータでドライバの削除指示を行う場合も考えられる。図17、18では、デバイスを共有設定しているコンピュータ（本発明の情報処理装置）でドライバの削除指示を行う場合のフローチャートを示す。

【0055】まず、ステップS801、S802は、ステップS1001、S1002と同様の処理である。ステップS803で、インストールされているドライバについて全てチェックしたかどうか判定する。全てのドライバについてチェックした場合は、この処理を終了する。

【0056】全てのドライバをチェックしていない場合、ステップS804に移行し、ドライバに対応するデバイスがローカルデバイスか、ネットワークで共有されたデバイスかを判定する。ローカルデバイスでなかった場合は、ステップS803に戻り、次のドライバをチェックする。

【0057】ローカルデバイスであった場合には、ステップS805に移行し、ドライバに対応するデバイスが存在するか否かを判定する。判定方法は、ステップS1005と同様である。ドライバに対応するデバイスが存在する場合は、ステップS811に移行する。

【0058】ドライバに対応するデバイスが存在しない場合、ステップS806に移行し、そのドライバが共有設定を行っていたか否かを判定する。共有設定していなかった場合は、ステップS808に移行する。

【0059】共有設定していた場合、ステップS807に移行し、対象デバイスが使用できなくなったことを他のコンピュータにネットワークを通じて通知し、ステップS808に移行する。図19は、ドライバの削除を通知する情報構造体の一例であり、デバイス種類、ドライバ名、バージョン情報、出力ポート等の情報を持っている。

【0060】上記の処理においては、デバイスを共有設定しているコンピュータ（本発明の情報処理装置）では、各PC（外部情報処理装置）の周辺機器からドライバのセットアップ状況情報を取得する。周辺機器のドライバとは、プリンタ、スキャナ、デジタルカメラ、FAX等のドライバのことである。セットアップ状況情報の取得方法としては、例えば、各PC上で自機にインストールされているドライバ情報を取得するモジュールを起動させることで情報を要求して収集して、本情報処理装置が外部情報処理装置（各PC）からそれらの情報をネットワークを通じて通信することが考えられる。

【0061】図21は、セットアップされているドライバ情報構造体の一例を示す図である。PC名称、IPアドレス、OS種類、ユーザ名、デバイスドライバ数、及び各デバイスドライバの情報から構成される。各デバイスドライバ情報はプリンタやスキャナといったデバイス種類、ドライバ名、バージョン情報、出力ポート、共有名、ドライバ情報アドレスで構成される。例えば、1番目に登録されているプリンタはドライバ名がLASER-830、バージョンが1.00.00、出力ポートがローカル、共有名がLASER-830、ドライバ情報アドレスが0x10000となっている。この情報を各PC毎に記録する。

【0062】上記のようにセットアップ情報を各PC間で送受信することにより、各PCにおいて共有設定されているデバイスのドライバを認識することができる。この情報を用いて、他のPCの使用できなくなったローカルデバイスを共有設定しているPC（本発明の情報処理装置）は、共有設定されている当該ローカルデバイスのドライバを有するPCに対して、当該ローカルデバイスが使用できなくなったことを通知する。当該通知を受けた各PCは、当該デバイスに対応するドライバの指定を含む通知を受けると、図18に示すように、当該デバイスに対応するドライバの削除処理を行う。また、ドライバの削除の通知を他のPCの使用できなくなったローカルデバイスを共有設定しているPCからドライバの削除を通知されるPCに送信する前に、当該他のPCの使用できなくなったローカルデバイスを共有設定しているPCにおいて、ドライバ削除通知の送信処理をOSに行わせるか否かを指示入力する指示入力画面（指示手段）、或いは、ドライバ削除通知の通知先のPCにおいて、ドライバを削除させるか否かのメッセージ、並びに、ユーザが削除通知の送信処理の実行を指示可能な指示画面を表示してもよい。また、一回のドライバ削除の通知によ

り、通知先のPCに複数のドライバの削除を行わせるようにしてもよい。

【0063】なお、本実施の形態では、対象のローカルデバイスのドライバをセットアップしているPCを判別し、そのPCに対してのみ共有の解除による削除指示を行ったがこれに限るものではなく、ネットワーク上のすべてのPCに対して、削除指示を行い、各PCにおいて、自身でそのドライバがセットアップされている場合に削除処理を行わせるようにしてもよい。

【0064】ステップS808では、対象ドライバが削除禁止設定されているか否かを判定する。判定方法は、ステップS1005の場合と同様である。削除禁止設定されている場合は、ステップS803に戻り、次のドライバをチェックする。

【0065】削除禁止設定されていない場合、ステップS809に移行し、削除を実行するか否かを判断する。判断方法は、ステップS1008と同様である。ここで、ドライバの削除がキャンセルされた場合は、ステップS803に戻り、次のドライバをチェックする。

【0066】ドライバの削除実行が指示された場合、ステップS810に移行し、ドライバの削除処理を実行し、削除処理を終了した後にステップS803に戻り、次のドライバをチェックする。

【0067】ステップS811では、対象ドライバが共有設定を解除したか否かを判断する。ここで、対象ドライバが共有設定を解除されていない場合は、ステップS803に戻り、次のドライバをチェックする。

【0068】対象ドライバが共有設定を解除していた場合、ステップS812に移行し、対象デバイスが使用できなくなったことを他のコンピュータにネットワークを通じて通知する。

【0069】一方、上述したようなドライバ削除通知が通知されるコンピュータでは、図18に示すように、まず、ステップS901で、削除通知を受けたか否かを判断する。通知を受けていない場合は、処理を終了する。

【0070】通知を受けた場合、ステップS902に移行し、ドライバ削除通知に対応するドライバがあるか否かを判定する。対応するドライバがない場合は、処理を終了する。

【0071】対応するドライバがある場合、ステップS903に移行し、対象ドライバが削除禁止設定されているか否かを判定する。判定方法は、ステップS1005と同様である。ドライバの削除禁止設定されている場合は、処理を終了する。

【0072】削除禁止設定されていない場合、ステップS904に移行し、メッセージを表示するか否かを判断する。メッセージ表示設定がされていない場合は、ステップS906に移行する。メッセージ表示設定されている場合、ステップS905に移行し、削除を行うか否かの確認メッセージを表示する。

【0073】そして、ステップS905で削除指示が入力指示されなかった場合は、処理を終了する。ドライバの削除指示が入力指示された場合、ステップS906に移行し、対象ドライバを削除して処理を終了する。

【0074】以上述べたように、ネットワーク上で共有されているPC及び周辺機器が表示されている画面において、ネットワーク上で共有されている周辺機器が削除或いは共有設定が解除される等して使用できなくなった場合に、不必要になったドライバを簡単な操作でアンインストールできる。また、共有設定していた周辺機器のドライバを削除したり、共有設定を解除した場合に他のコンピュータに通知することで不要なドライバを削除したりすることができる。これらの機能によって、ネットワーク上での作業効率を大幅に向上させることができる。

【0075】以下、図20～31を参照して、周辺機器のドライバのバージョンアップを行う場合について説明する。図20は、ドライバのバージョンアップを行う処理動作を示すフローチャートである。まず、ステップS2001で、ネットワーク上の全てのPC及び周辺機器の接続状況情報を取得し、その情報を記憶する。

【0076】次に、ステップS2002で、各PCの周辺機器のドライバにセットアップ状況情報を取得する。周辺機器のドライバとは、プリンタ、スキャナ、デジタルカメラ、FAX等のドライバのことである。セットアップ状況情報の取得方法としては、例えば、各PC上で自機にインストールされているドライバ情報を取得するモジュールを起動させることで情報を収集して、それらの情報をネットワークを通じて通信することが考えられる。

【0077】図21は、セットアップされているドライバ情報構造体の一例を示す図である。PC名称、IPアドレス、OS種類、ユーザ名、デバイスドライバ数、及び各デバイスドライバの情報から構成される。各デバイスドライバ情報はプリンタやスキャナといったデバイス種類、ドライバ名、バージョン情報、出力ポート、共有名、ドライバ情報アドレスで構成される。例えば、1番目に登録されているプリンタはドライバ名がLASER-830、バージョンが1.00.00、出力ポートがローカル、共有名がLASER-830、ドライバ情報アドレスが0x10000となっている。この情報を各PC毎に記録する。

【0078】ステップS2003で、取得した情報を基にネットワーク上の全てのPC及び周辺機器の接続状況情報を表示する。この画面表示については、図12により既述したので、その詳細な説明を省略する。

【0079】図22は、ネットワーク上の各PCのドライバセットアップ状況を表示した画面の一例を示す図であり、各PCから取得したドライバ情報構造体を基に表示される。例えば、同図では、私のPCにインストールされているドライバは、プリンタが6個、スキャナが2

個ある。この中で、例えば、「INKJET-10V」というプリンタは、ドライバ名が「INKJET-10V」で、バージョン番号が「1.00.00」で、出力ポート名が「¥¥NOTEPC01¥¥INKJET-10V」で、名称「INKJET-10V」で共有設定されているネットワーク上で共有されているプリンタドライバであることを示している。

【0080】図20のフローチャートに説明を戻すと、ステップS2004で、周辺機器のドライバが更新されたか否かを判定する。ここで、ドライバが更新されていない場合は、処理を終了する。ドライバが更新されている場合、ステップS2005に移行する。ドライバの更新方法としては、各PC上でドライバを更新して、サーバ機器となるPCにそのドライバ情報を登録することが考えられる。または、ユーザが最新のドライバを各メーカーのサイトからダウンロードして登録することも考えられる。さらに、ネットワーク上の特定の場所を指定しておいて、ドライバのバージョンチェックを行うこともできる。図23はドライバのバージョンチェック設定を行うための画面の一例である。このように、特定のURLを設定しておくことで、定期的にドライバのバージョンチェックを行って、ドライバの更新があった場合には自動的に取得するように設定できる。

【0081】ステップS2005では、更新されたドライバが最新バージョンであるか否かを判定する。そして、最新バージョンでなかった場合は、更新する通知をする必要がないので処理を終了する。最新バージョンであった場合、ステップS2006に移行する。ドライバのバージョンの判定方法としては、サーバ機器で管理している各プリンタのドライバ情報を基に行うことが考えられる。図24はそれぞれのドライバ毎にバージョン管理されている構造体の一例であり、プリンタやスキャナといったデバイス種類、デバイスのドライバ名、管理されているバージョン情報数、ドライババージョン情報、及びそれに対応するドライバ情報から構成される。管理しているバージョン情報と更新されたドライバのバージョン情報とを比較することで判定する。

【0082】ステップS2006では、更新されたドライバを使用しているPCがあるか否かを判定する。判定は、各PCから取得したデバイスドライバ情報構造体を用いて行う。ここで、対象PCがなかった場合は、処理を終了する。対象PCがあった場合、対象PCリストを表示する。図25はドライバの更新対象PCを選択するための画面の一例であり、ここで対象PCを選択して「OK」を選択すると、ステップS2007に移行し、選択されたPCに通知が行われる。また、「対象PC全てを選択」を選択すると、全ての対象PCが選択される。

【0083】ステップS2007では、選択された対象PCに対してドライバが更新されたことを通知する。図26はデバイス更新通知の情報構造体の一例であり、デ

バイス種類、ドライバ名、バージョン情報、出力ポートから構成される。

【0084】そして、ステップS2008に移行し、通知したPCからドライバ情報の取得要求があるか否かを判定する。要求がない場合は、或いはドライバ情報を送信する必要がないとクライアントPCから応答があった場合は、ステップS2010に移行する。要求がある場合、ステップS2009に移行し、要求があったドライバ情報をクライアントPCに送信する。

10 【0085】ステップS2010では、通知した全てのPCから応答があったかどうか判定する。全てのPCから応答があった場合は、処理を終了する。全てのPCから応答がまだない場合、ステップS2008に戻る。

【0086】一方、図27は、クライアント機器における処理動作を示すフローチャートである。まず、ステップS901で、サーバ機器からドライバの更新通知があったか否かを判定する。ここで、通知がない場合は、処理を終了する。

20 【0087】通知があった場合、ステップS902に移行し、ドライバ更新をするか否かを判定する。例えば、ドライバ更新の通知があったことをメッセージ表示してユーザが指示する方法が考えられる。図28はドライバが更新されたことを通知するメッセージの一例である。ここで、「いいえ」が選択された場合は、サーバ機器に対して対象ドライバ情報が必要でないことを通知して、処理が終了される。「はい」が選択された場合、ステップS903に移行し、サーバ機器に対して対象ドライバ情報を要求する。

30 【0088】次に、ステップS904で、ドライバ情報が取得できたか否かを判定する。取得できない場合は、ステップS903に戻り、サーバ機器に対して再度取得要求を送信する。ドライバ情報を取得できた場合、ステップS905に移行し、ドライバの更新処理を行う。さらに、ステップS906で最新のドライバインストール状況をサーバ機器に通知して処理を終了する。

40 【0089】図20及び27に示したフローチャートでは、ドライバの更新処理がされた場合に、通知を行うPCの選択をサーバ機器となるPCで行っているが、クライアント機器となるPCで自機にインストールされているドライバ状況から判定してドライバの更新処理を行う場合も考えられる。図29～31は、クライアント機器となるPCで自機にインストールされているドライバ状況から判定して、ドライバの更新処理を行う場合のフローチャートである。

【0090】図29のステップS1201～ステップS1205は、ステップS2001～ステップS2005と同じである。次に、ステップS1206で、ドライバが更新されたことを全てのPCに通知して処理を終了する。通知する情報構造体も、図26に示した通りである。

【0091】そして、図30に示すフローチャートに従って、クライアント機器からの要求を処理する。まず、ステップS1301で、ドライバ情報の要求があるか否かを判定する。要求がない場合は、処理を終了する。

【0092】要求があった場合、ステップS1302に移行し、管理しているドライバ情報から要求があったドライバ情報を読み出す。そして、ステップS1303で、そのドライバ情報を要求があったクライアント機器に送信する。

【0093】一方、クライアント機器においては、図31に示すフローチャートに従って処理が行われる。このフローチャートは、図27に示すフローチャートとほぼ同じであり、ステップS1401、ステップS1403～ステップS1407は図27のステップS901～ステップS906で説明したとおりであるが、ステップS1402の判定があることが異なる。

【0094】このステップS1402では、ドライバ更新通知をサーバ機器から受けた場合に、ドライバ更新通知構造体から必要データを読み出して、また、自機にインストールされているドライバ情報と比較して、通知された対象ドライバがあるか否かを判定する。そして、対象ドライバがなかった場合は、この通知を無視して処理を終了する。対象ドライバがあった場合、図27に示すフローチャートと同様の処理を行う。

【0095】以上述べたように、ネットワーク上で共有されているPC及び周辺機器が表示されている画面において、周辺機器のドライバが更新された場合に、簡単な操作でドライバの更新処理を行うことができる。また、ドライバの更新情報のみを各PCに通知することで、ドライバの更新処理を行うかどうか判定できるので、ネットワーク上のトラフィックを最小限に抑えることができる。これらの機能によって、ネットワーク上での作業効率を大幅に向上させることができる。なお、上述の各実施形態において、ドライバ削除手段は、OSのアンインストールモジュールとして提供される場合もある。この場合でも、ドライバを削除するようOSを制御することにより、ドライバの削除処理を行うアプリケーションとして実装されるアンインストールプログラム（ドライバ削除手段）も本発明の範疇となる。

【0096】（その他の実施の形態）上述した実施の形態の機能を実現するべく各種のデバイスを動作させるように、該各種デバイスと接続された装置或いはシステム内のコンピュータに対し、上記実施の形態の機能を実現するためのソフトウェアのプログラムコードを供給し、そのシステム或いは装置のコンピュータ（CPU或いはMPU）に格納されたプログラムに従って上記各種デバイスを動作させることによって実施したものも、本発明の範疇に含まれる。

【0097】また、この場合、上記ソフトウェアのプログラムコード自体が上述した実施の形態の機能を実現す

ることになり、そのプログラムコード自体、及びそのプログラムコードをコンピュータに供給するための手段、例えばかかるプログラムコードを格納した記録媒体は本発明を構成する。かかるプログラムコードを記憶する記録媒体としては、例えばフレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、磁気テープ、不揮発性のメモ리카ード、ROM等を用いることができる。

【0098】また、コンピュータが供給されたプログラムコードを実行することにより、上述の実施の形態の機能が実現されるだけでなく、そのプログラムコードがコンピュータにおいて稼働しているOS（オペレーティングシステム）或いは他のアプリケーションソフト等と共同して上述の実施の形態の機能が実現される場合にもかかるプログラムコードは本発明の実施の形態に含まれることはいうまでもない。

【0099】さらに、供給されたプログラムコードがコンピュータの機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに格納された後、そのプログラムコードの指示に基づいてその機能拡張ボードや機能拡張ユニットに備わるCPU等が実際の処理の一部又は全部を行い、その処理によって上述した実施の形態の機能が実現される場合にも本発明に含まれることはいうまでもない。

【0100】なお、上記実施の形態において示した各部の形状及び構造は、何れも本発明を実施するにあたっての具体化のほんの一例を示したものに過ぎず、これらによって本発明の技術的範囲が限定的に解釈されてはならないものである。すなわち、本発明はその精神、又はその主要な特徴から逸脱することなく、様々な形で実施することができる。

【0101】

【発明の効果】以上説明したように本発明によれば、ネットワーク上で共有されている周辺機器が削除されたり、共有設定が解除されたりして使用できなくなった場合に、不必要になったドライバを簡単な操作でアンインストールできる。また、共有設定していた周辺機器のドライバを削除したり、共有設定を解除した場合に他のコンピュータに通知することで、不要なドライバを削除したりすることができる。これらの機能によって、ネットワーク上での作業効率を大幅に向上させることができる。

【図面の簡単な説明】

【図1】本実施の形態の情報処理装置の概略構成を示すブロック図である。

【図2】ドライバのセットアップを行う処理動作を示すフローチャートである。

【図3】ドライバ情報構造体の一例を示す図である。

【図4】ネットワーク上のPC及び周辺機器を表示する画面の一例を示す図である。

【図5】ネットワーク上の各PCのドライバセットアップ状況を表示する画面の一例を示す図である。

【図6】セットアップするドライバを選択するための画面の一例を示す図である。

【図7】ドライバをセットアップするPCを選択するための画面の一例を示す図である。

【図8】ドライバセットアップ指示構造体の一例を示す図である。

【図9】ドライバのセットアップ確認印刷の一例を示す図である。

【図10】ドライバのチェック、さらには削除を行う処理動作を示すフローチャートである。

【図11】ドライバ情報構造体の一例を示す図である。

【図12】ネットワーク上のPC及び周辺機器を表示した画面の一例を示す図である。

【図13】アイコン303iで表されるプリンタの共有設定が解除された、或いはドライバが削除された場合に表示される画面の一例を示す図である。

【図14】チェックするデバイスを選択するための画面の一例を示す図である。

【図15】ドライバの削除禁止設定を行うための画面の一例を示す図である。

【図16】削除するドライバがあった場合に表示されるメッセージの一例を示す図である。

【図17】ドライバのチェック、さらには削除を行う処理動作を示すフローチャートである。

【図18】ドライバ削除通知が通知されたときの処理動作を示すフローチャートである。

【図19】ドライバの削除を通知する情報構造体の一例を示す図である。

【図20】ドライバのバージョンアップを行う処理動作を示すフローチャートである。

【図21】ドライバ情報構造体の一例を示す図である。

【図22】ネットワーク上の各PCのドライバセットアップ状況を表示した画面の一例を示す図である。

【図23】ドライバのバージョンチェック設定を行うための画面の一例を示す図である。

【図24】ドライバ毎にバージョン管理されている構造体の一例を示す図である。

【図25】ドライバの更新対象PCを選択するための画面の一例を示す図である。

【図26】デバイス更新通知の情報構造体の一例を示す図である。

【図27】ドライバのバージョンアップを行う処理動作を示すフローチャートである。

【図28】ドライバが更新されたことを通知するメッセージの一例を示す図である。

【図29】ドライバのバージョンアップを行う処理動作を示すフローチャートである。

【図30】ドライバのバージョンアップを行う処理動作を示すフローチャートである。

【図31】ドライバのバージョンアップを行う処理動作を示すフローチャートである。

【符号の説明】

- 1 システムバス
- 2 CPU
- 3 プログラムメモリ
- 4 通信制御部
- 5 通信ポート
- 6 通信回線
- 7 ネットワーク上の他の装置
- 8 外部記憶装置制御部
- 9 フレキシブルディスク
- 10 ハードディスク
- 11 入力制御部
- 12 キーボード
- 13 マウス
- 14 ビデオイメージメモリ
- 15 表示出力制御部
- 16 CRT
- 17 プリンタ制御部
- 18 プリンタ
- 1A 画像読み取り装置制御部
- 1B 画像読み取り装置

【図7】

ドライバのセットアップ

ドライバをセットアップするPCを選択してください。

ドメイン名(D):

PC名(P):

ドメイン2

SNPC01
SNPC02
SNPC03
SNPC04
SNPC05
SNPC06
SNPC07

戻る(B) 完了 キャンセル

【図8】

デバイス種類	プリンタ
ドライバ名	Printer 2000
バージョン情報	1.00.00
出力ポート	\\Vsnpcw2\VPm2000
セットアップ情報	0

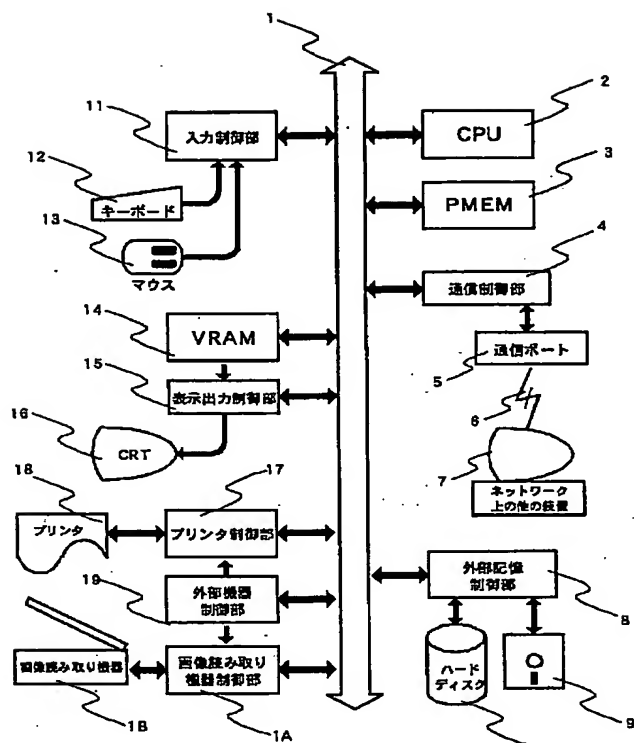
【図19】

デバイス種類	プリンタ
ドライバ名	Printer 2000
バージョン情報	1.00.00
出力ポート	\\Vsnpcw2\VPm2000

【図26】

デバイス種類	プリンタ
ドライバ名	Printer 2000
バージョン情報	1.20.00
出力ポート	\\Vsnpcw2\VPm2000

【図1】



【図3】

PC名称	SNPC00
IPアドレス	100.100.100.1
OS種類	Windows98
ユーザ名	suzuki
ドライバ数	N
1	デバイス種類 プリンタ ドライバ名 Printer 2000 バージョン情報 1.00.00 出力ポート ローカル 共有情報 ON
2	デバイス種類 スキャナ ドライバ名 Scanner 2000 バージョン情報 1.00.00 出力ポート ローカル 共有情報 ON
	...
N	デバイス種類 プリンタ ドライバ名 Printer 2001 バージョン情報 1.00.00 出力ポート ネットワーク 共有情報 OFF

【図23】

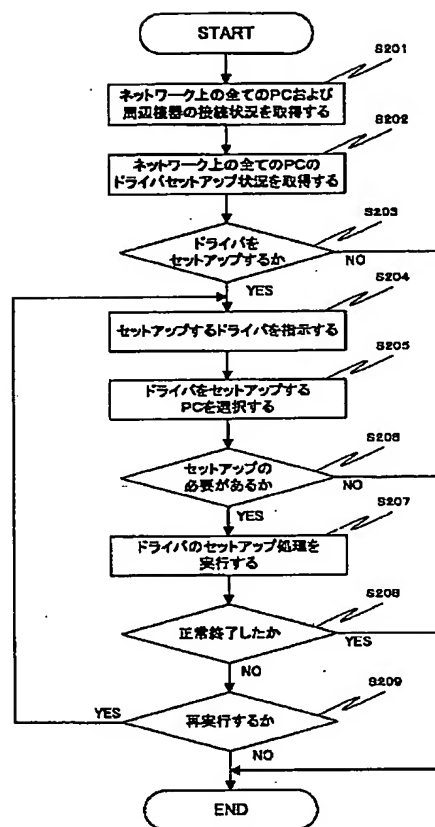
ドライババージョンチェック設定

ドライバのバージョンチェックを行うURLを設定してください。

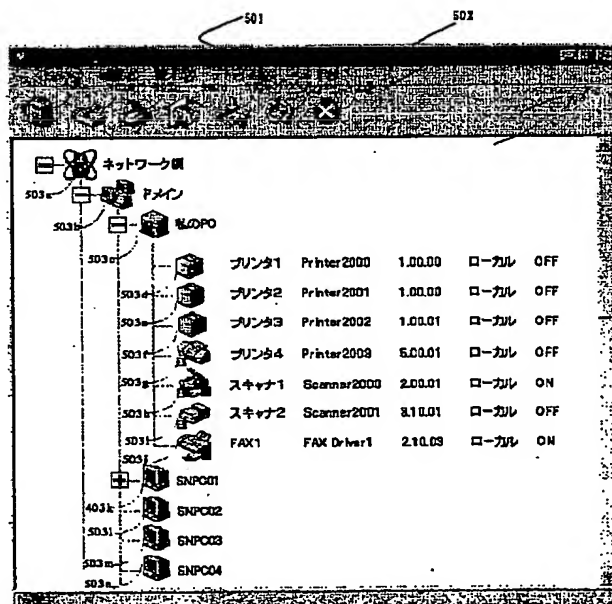
URL(U):

OK キャンセル

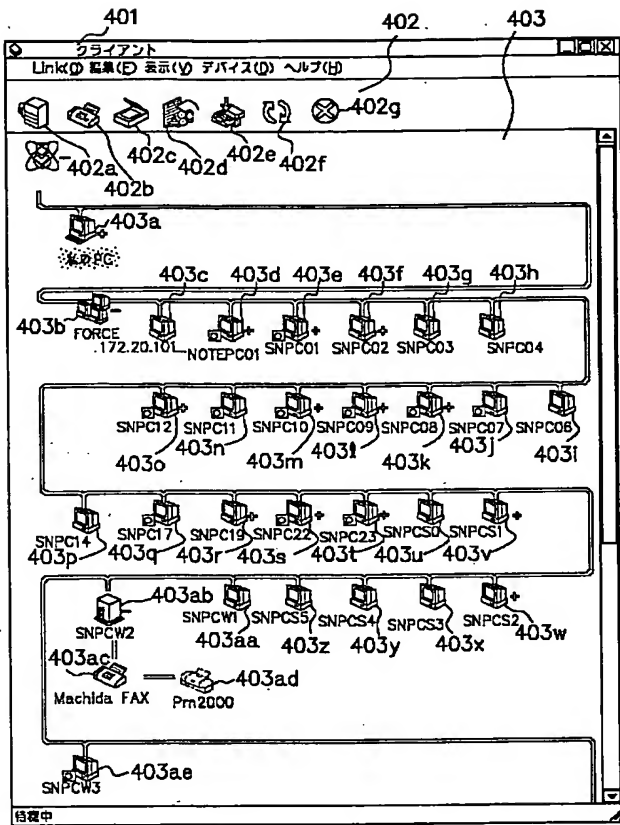
【図2】



【図5】



【図4】



【図6】

ドライバのセットアップ

プリンタの製造元とモデルを選んでください。プリンタにインストールディスクが付いている場合は、[ディスク使用]を押して下さい。プリンタが一覧にない場合は、プリンタのマニュアルを参照して互換性のあるプリンタを選んでください。

製造元 (M):
A社
B社

プリンタ (P):
A社 LaserWriter JNTX-J
A社 LaserWriter Pro 400
A社 LaserWriter Pro 405
A社 LaserWriter Select 610

ディスク使用 (H)

次へ> キャンセル

【図16】

削除確認

? "INKJET-10V" ドライバに対応するプリンタはありません。ドライバを削除しますか?

はい (Y) いいえ (N)

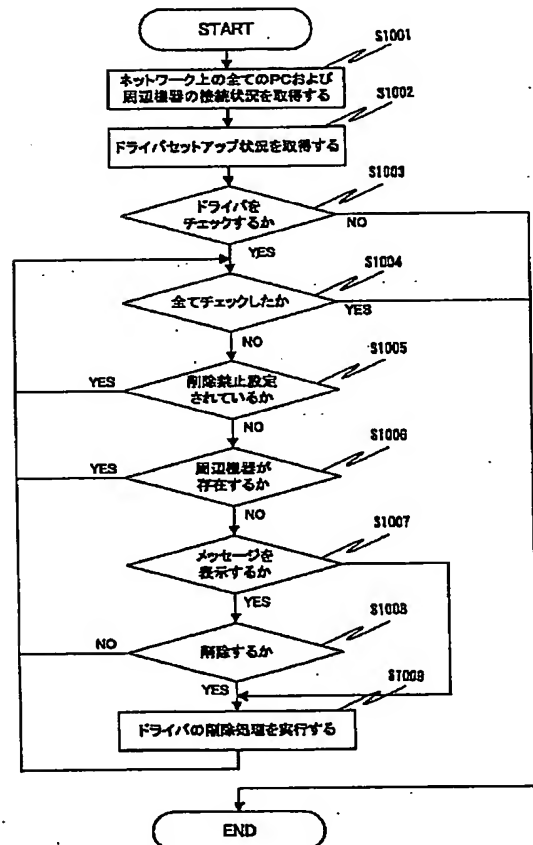
【図9】

プリンタの印字テスト

Printer 2000 のインストールが完了しました。
PC名、プリンタドライバおよびポートの設定は以下の通りです。

日時: 2000年6月12日
PC名: SNPC01
ドライバ名: Printer2000
バージョン: 1.00.00
出力ポート: \\\$SNPCW2\$Pm2000

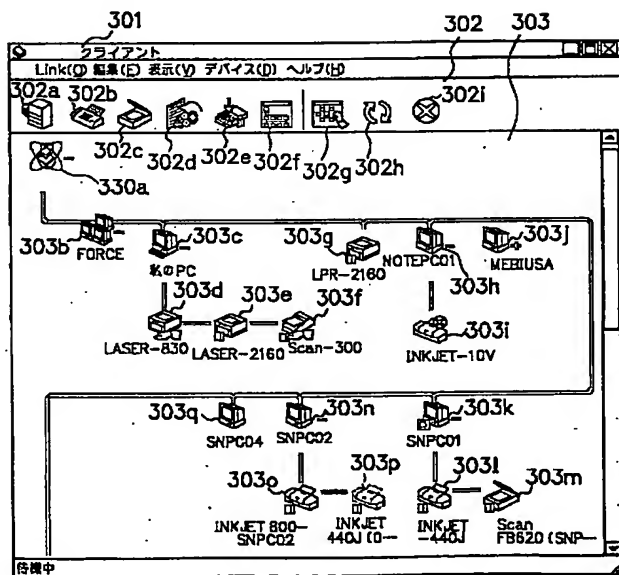
【図10】



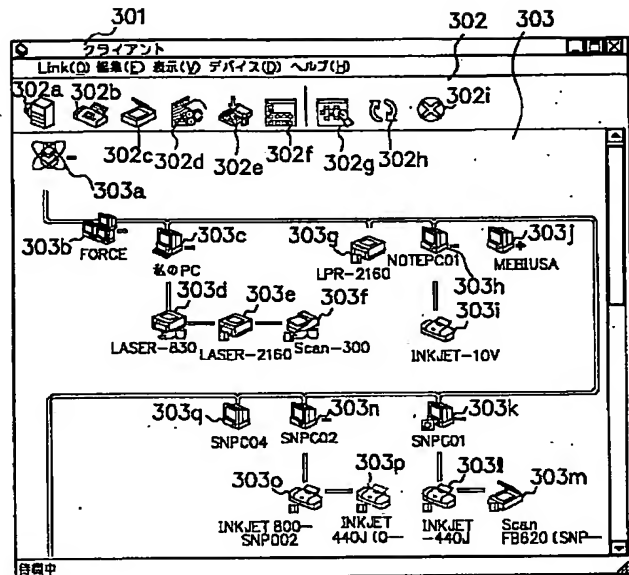
【図11】

PC名称		SNPC00
IPアドレス		100.100.100.1
OS種類		Windows98
ユーザ名		suzuki
ドライバ数		8
1	デバイス種類	プリンタ
	ドライバ名	LASER-830
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	ON
2	デバイス種類	プリンタ
	ドライバ名	LBP-2160
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	OFF
3	デバイス種類	プリンタ
	ドライバ名	Scanner2000
	バージョン情報	1.00.00
	出力ポート	100.100.100.123
	共有情報	OFF
4	デバイス種類	プリンタ
	ドライバ名	INKJET-10V
	バージョン情報	1.00.00
	出力ポート	***NOTEPC01*INKJET-10V
	共有情報	OFF
7	デバイス種類	スキャナ
	ドライバ名	Scan_300
	バージョン情報	1.00.00
	出力ポート	ローカル
	共有情報	OFF
8	デバイス種類	スキャナ
	ドライバ名	Scan_FB620
	バージョン情報	1.00.00
	出力ポート	***snpc01*FB620
	共有情報	OFF

【図13】



【図12】



【図14】

ドライバのチェック

デバイスの接続状態とインストールされているドライバをチェックして、不要なドライバを削除します。
実行する場合は、チェックするデバイスを選択して、
[OK] ボタンを選択してください。
キャンセルする場合は、[キャンセル] ボタンを選択してください。

デバイス種類 (D):

☐ プリンタ
☐ スキャナ
☐ FAX

☐ 確認メッセージを表示する。

OK キャンセル

【図15】

ドライバの削除禁止設定

ドライバの削除を自動的に行わないドライバを設定してください。

禁止するドライバ (R)

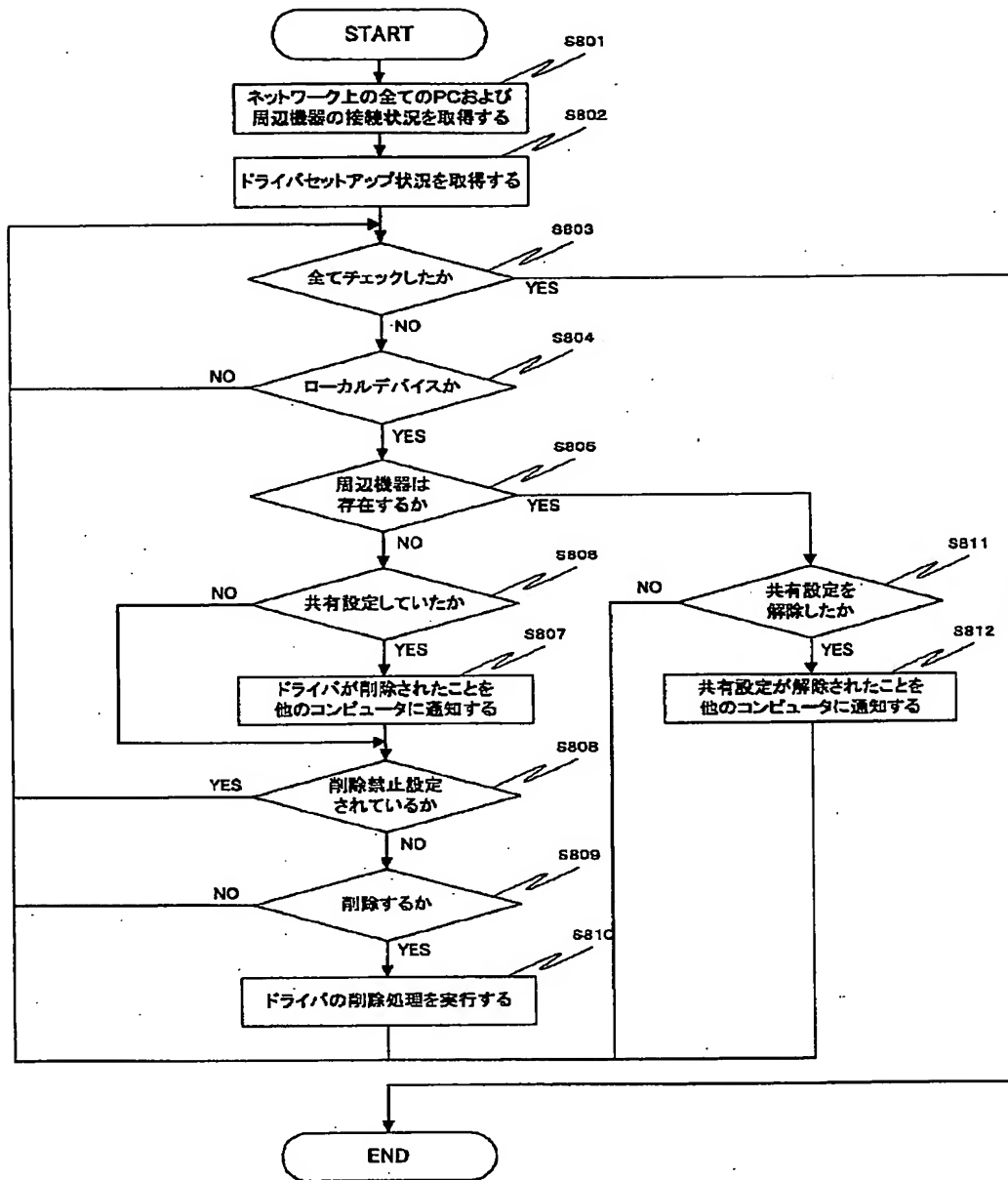
参照 (S) 追加 (A)

禁止するドライバー名 (L)

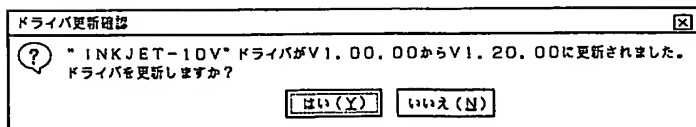
***snpc01*INKJET-440J 削除 (R)

OK キャンセル

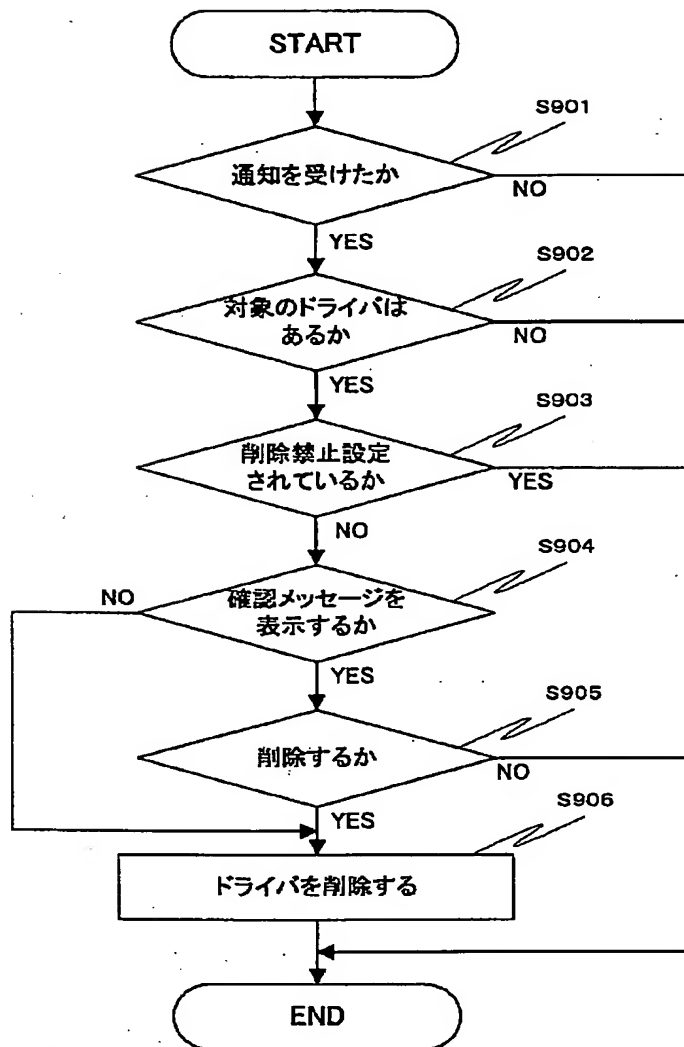
【図17】



【図28】



【図18】



【図21】

PC名称	SNPC00
IPアドレス	100.100.100.1
OS種類	Windows98
ユーザ名	suzuki
ドライバ数	8
1	デバイス種類 プリンタ ドライバ名 LASER-830 バージョン情報 1.00.00 出力ポート ローカル 共有名 LASER-830 ドライバ情報アドレス 0x10000
2	デバイス種類 プリンタ ドライバ名 LASER-2160 バージョン情報 1.00.00 出力ポート ローカル 共有名 0 ドライバ情報アドレス 0x20000
3	デバイス種類 プリンタ ドライバ名 INKJET-10V バージョン情報 1.00.00 出力ポート *NOTPC01*INKJET-10V 共有名 INKJET-10V ドライバ情報アドレス 0x30000
	...
8	デバイス種類 スキャナ ドライバ名 Scan FB620 バージョン情報 1.00.00 出力ポート *snpc01*FB620 共有名 FB620 ドライバ情報アドレス 0x40000
	ドライバ情報

【図24】

デバイス種類	プリンタ
ドライバ名	LASER-830
バージョン情報数	N
1	バージョン情報 1.00.00 ドライバ情報アドレス 0x10000
2	バージョン情報 1.20.00 ドライバ情報アドレス 0x20000
	...
N	バージョン情報 3.10.00 ドライバ情報アドレス 0x80000
	ドライバ情報

【図25】

ドライバのセットアップ

ドライバ更新通知を行うPCを選択してください。

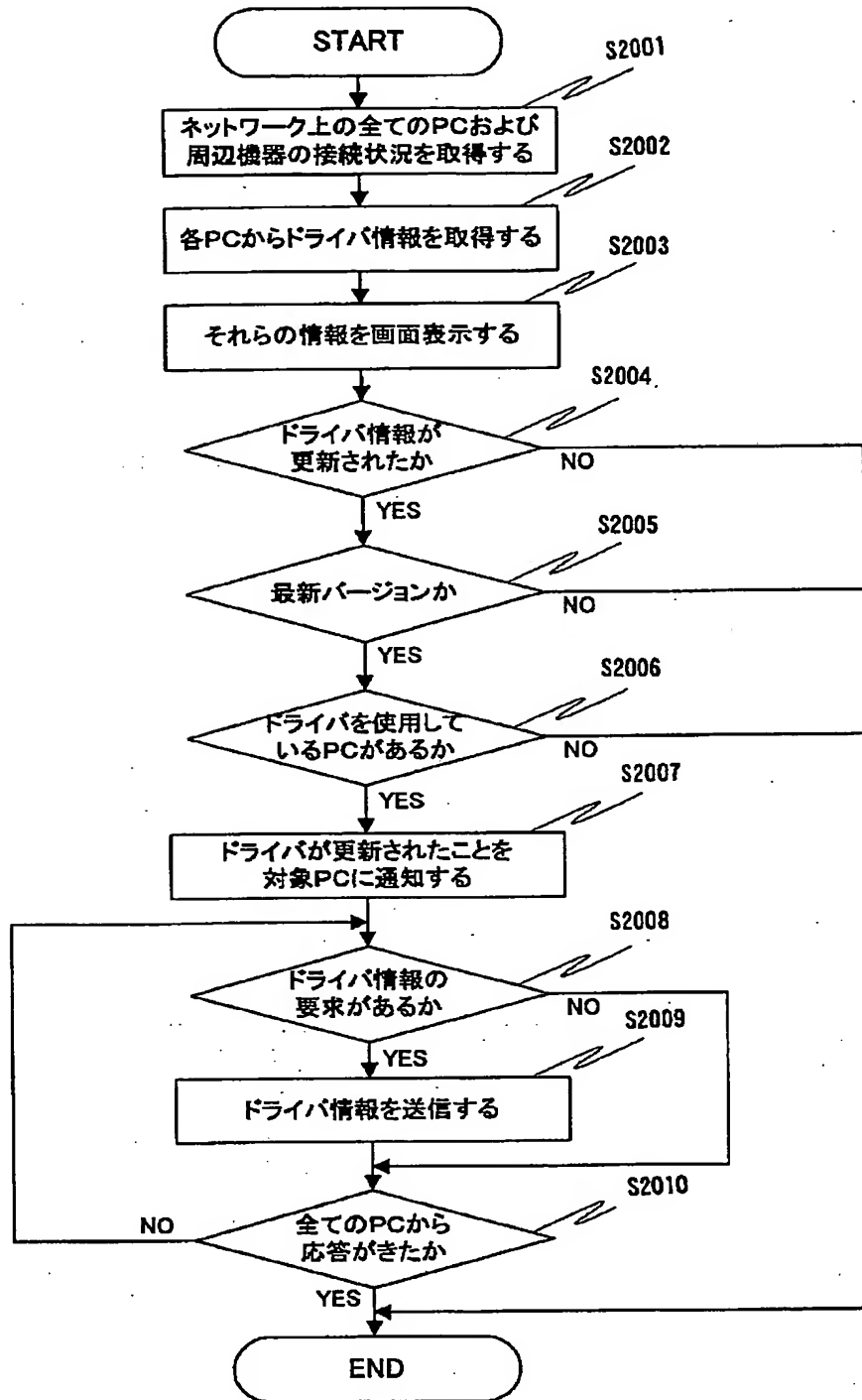
ドメイン名 (D): PC名 (P):

ドメイン2	SNPC01 1.00.00
	SNPC02 1.00.00
	SNPC03 1.00.00
	SNPC04 1.00.00
	SNPC05 1.00.00
	SNPC06 1.00.00
	SNPC07 1.00.00

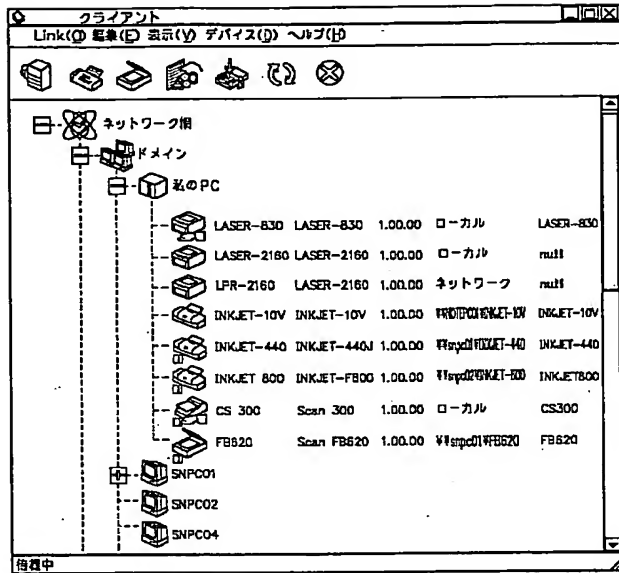
対象PC全てを選択

OK キャンセル

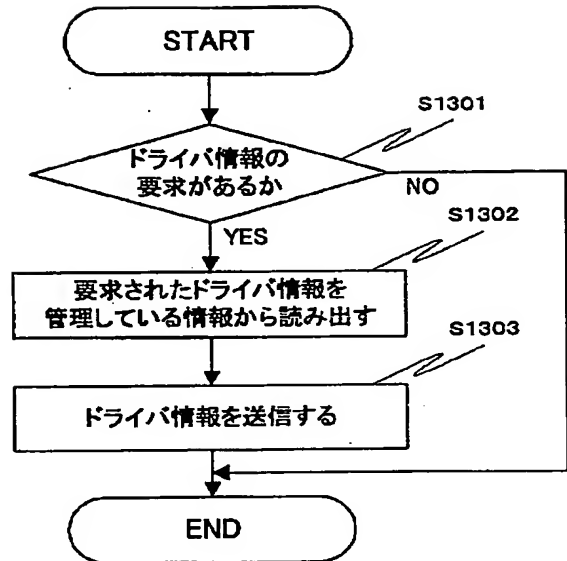
【図20】



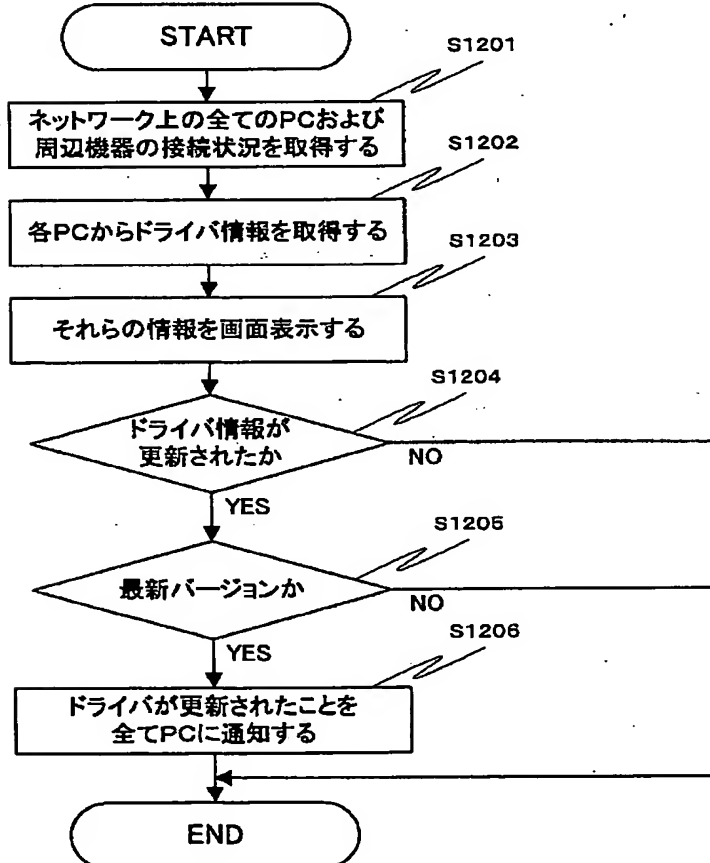
【図22】



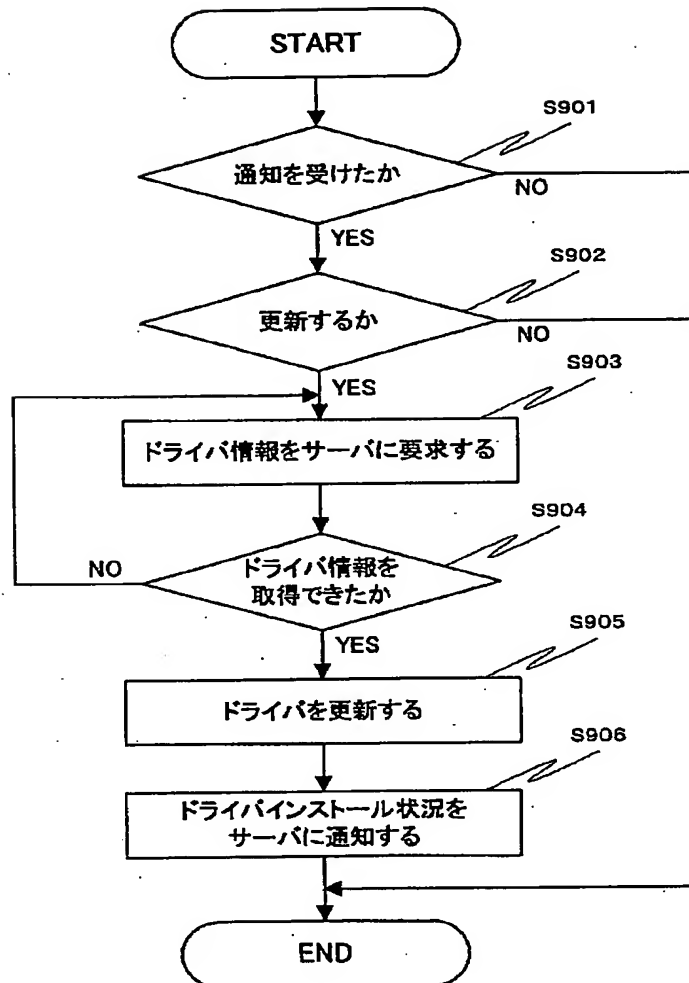
【図30】



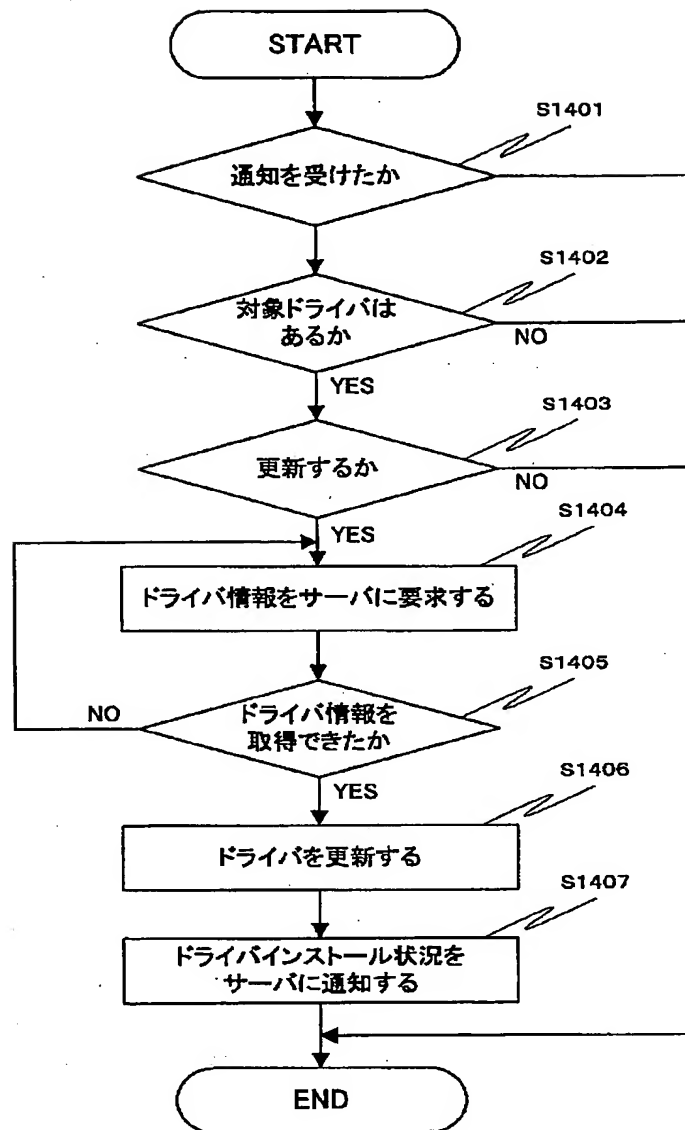
【図29】



【図27】



【図 31】



【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】平成15年3月14日(2003.3.14)

【公開番号】特開2002-251366(P2002-251366A)

【公開日】平成14年9月6日(2002.9.6)

【年通号数】公開特許公報14-2514

【出願番号】特願2001-355016(P2001-355016)

【国際特許分類第7版】

G06F 13/10 320
9/445

【F I】

G06F 13/10 320 A
9/06 610 B
610 D
610 C
610 L

【手続補正書】

【提出日】平成14年12月10日(2002.12.10)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】 周辺機器と通信する情報処理装置であって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、

前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段とを備えたことを特徴とする情報処理装置。

【請求項2】 前記判定手段は、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在するか否かを判定することを特徴とする請求項1に記載の情報処理装置。

【請求項3】 前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手段を備えたことを特徴とする請求項1に記載の情報処理装置。

【請求項4】 前記ドライバ削除手段によるドライバの削除の実行を指示する指示手段を備え、前記ドライバ削除手段は、前記指示手段による一回の指示に応じて、複数のドライバの削除を実行することを特徴とする請求項3に記載の情報処理装置。

【請求項5】 前記ドライバ削除手段により前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項3に記載

載の情報処理装置。

【請求項6】 所望のドライバの削除の禁止設定を行う削除禁止設定手段を備えたことを特徴とする請求項3に記載の情報処理装置。

【請求項7】 前記情報処理装置は外部情報処理装置と通信するものであり、

前記判定手段が、前記ドライバに対応する周辺機器が存在しないと判断し、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドライバ削除通知を行うドライバ削除通知手段をさらに備えたことを特徴とする請求項1に記載の情報処理装置。

【請求項8】 周辺機器と通信する情報処理装置における情報処理方法であって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、

前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段とを有することを特徴とする情報処理方法。

【請求項9】 前記判定手段では、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在するか否かを判定することを特徴とする請求項8に記載の情報処理方法。

【請求項10】 前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手段を有することを特徴とする請求項8に記載の情報処理方法。

【請求項11】 前記ドライバ削除手段において、ドライバの削除の実行を指示する指示手段を行い、前記ドライバ削除手段では、前記指示手段による一回の指示に応じて、複数のドライバの削除を実行することを特徴とす

る請求項 10 に記載の情報処理方法。

【請求項 12】 前記ドライバ削除手順で前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認手順を有することを特徴とする請求項 10 に記載の情報処理方法。

【請求項 13】 所望のドライバの削除の禁止設定を行う削除禁止設定手順を有することを特徴とする請求項 10 に記載の情報処理方法。

【請求項 14】 前記情報処理装置は外部情報処理装置と通信するものであり、前記判定手順で前記ドライバに対応する周辺機器が存在しないと判断され、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドライバ削除通知を行うドライバ削除通知手順をさらに有することを特徴とする請求項 8 に記載の情報処理方法。

【請求項 15】 周辺機器と通信する情報処理装置において実行される制御プログラムであって、前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得処理と、前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定処理とを前記情報処理装置において実行させることを特徴とする制御プログラム。

【請求項 16】 周辺機器並びに外部情報処理装置と通信する情報処理装置であって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段とを備えたことを特徴とする情報処理装置。

【請求項 17】 前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段をさらに備え、前記ドライバ削除通知手段は、前記セットアップ情報に基づいて、ドライバの削除通知を行うことを特徴とする請求項 16 に記載の情報処理装置。

【請求項 18】 前記ドライバ削除通知手段によるドライバの削除通知の実行を指示する指示手段を備えたことを特徴とする請求項 16 に記載の情報処理装置。

【請求項 19】 前記ドライバ削除通知手段は、前記指示手段による一回の指示に応じて、前記外部情報処理装置に複数のドライバを削除させるドライバ削除通知を行うことを特徴とする請求項 18 に記載の情報処理装置。

【請求項 20】 前記ドライバ削除通知手段により前記ドライバの削除を通知する前に、削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項 16 に記載の情報処理装置。

【請求項 21】 所望のドライバの削除の禁止設定を行う削除禁止設定手段を備えたことを特徴とする請求項 16 に記載の情報処理装置。

【請求項 22】 前記判定手段が、前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うことを特徴とする請求項 16 に記載の情報処理装置。

【請求項 23】 周辺機器並びに外部情報処理装置と通信する情報処理装置における情報処理方法であって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手順と、前記判定手順により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手順とを有することを特徴とする情報処理方法。

【請求項 24】 前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手順をさらに有し、前記ドライバ削除通知手順は、前記セットアップ情報に基づいて、ドライバの削除通知を行うことを特徴とする請求項 23 に記載の情報処理方法。

【請求項 25】 前記ドライバ削除通知手段によるドライバの削除の通知を指示する指示手段を有することを特徴とする請求項 23 に記載の情報処理方法。

【請求項 26】 前記ドライバ削除通知手段は、前記指示手段による一回の指示に応じて、複数のドライバの削除を前記外部情報処理装置に実行させるドライバ削除通知を行うことを特徴とする請求項 25 に記載の情報処理方法。

【請求項 27】 前記ドライバ削除通知手段により前記ドライバ削除通知を行う前に、前記外部情報処理装置に削除を実行させるか否かをユーザに確認する確認手順を有することを特徴とする請求項 23 に記載の情報処理方法。

【請求項 28】 所望のドライバの削除の禁止設定を行う削除禁止設定手順を有することを特徴とする請求項 23 に記載の情報処理方法。

【請求項 29】 前記判定手順で前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うことを特徴とする請求項 23 に記載の情報処理方法。

【請求項 30】 周辺機器並びに外部情報処理装置と通信する情報処理装置において実行される制御プログラムであって、セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定処理と、前記判定処理により前記ドライバに対応する周辺機器が

存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知処理とを前記情報処理装置に実行させることを特徴とする制御プログラム。

【請求項31】 周辺装置並びに複数の外部情報処理装置と通信する情報処理装置において、前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手段と、前記認識手段により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手段とを備えたことを特徴とする情報処理装置。

【請求項32】 前記通知手段は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うことを特徴とする請求項31に記載の情報処理装置。

【請求項33】 前記認識手段は、前記ローカルデバイスが存在するか否かを認識する機能、及び、前記ローカルデバイスが共有設定されているか否かを認識する機能を備えたことを特徴とする請求項31に記載の情報処理装置。

【請求項34】 前記通知手段は、前記認識手段により前記ローカルデバイスが存在しないと認識された場合、又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うことを特徴とする請求項33に記載の情報処理装置。

【請求項35】 前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識することを特徴とする請求項31に記載の情報処理装置。

【請求項36】 周辺装置並びに複数の外部情報処理装置と通信する情報処理装置における情報処理方法において、前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手順と、

前記認識手順により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手順とを有することを特徴とする情報処理方法。

【請求項37】 前記通知手順は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うことを特徴とする請求項36に記載の情報処理方法。

【請求項38】 前記認識手順は、前記ローカルデバイスが存在するか否かを認識する手順、及び、前記ローカルデバイスが共有設定されているか否かを認識する手順を有することを特徴とする請求項36に記載の情報処理方法。

【請求項39】 前記通知手順は、前記認識手順により前記ローカルデバイスが存在しないと認識された場合、

又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うことを特徴とする請求項38に記載の情報処理方法。

【請求項40】 前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識することを特徴とする請求項36に記載の情報処理方法。

【請求項41】 周辺装置並びに複数の外部情報処理装置と通信する情報処理装置における制御プログラムにおいて、

前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識処理と、前記認識処理により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知処理とを前記情報処理装置に実行させることを特徴とする制御プログラム。

【請求項42】 外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手段と、前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えたことを特徴とする情報処理装置。

【請求項43】 前記ドライバ削除手段により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項42に記載の情報処理装置。

【請求項44】 所望のドライバの削除の禁止を設定する削除禁止設定手段を備えたことを特徴とする請求項42に記載の情報処理装置。

【請求項45】 前記削除禁止設定手段において、ドライバ削除の禁止が設定されている場合に、前記ドライバ削除手段がドライバを削除しないように制御する制御手段を備えたことを特徴とする請求項44に記載の情報処理装置。

【請求項46】 外部情報処理装置と通信して、セットアップ情報の収集を行う収集手段を備えたことを特徴とする請求項42に記載の情報処理装置。

【請求項47】 外部情報処理装置からの要求にตอบสนองして、セットアップ情報を返送することを特徴とする請求項45に記載の情報処理装置。

【請求項48】 前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手段をさらに備え、

前記ドライバ削除手段は、前記判断手段により指定されたドライバがセットアップされていない場合には、前記削除通知を無視することを特徴とする請求項42に記載の情報処理装置。

【請求項49】 外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手段と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手順とを有することを特徴とする情報処理方法。

【請求項 5 0】 前記ドライバ削除手順により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認手順を有することを特徴とする請求項 4 9 に記載の情報処理方法。

【請求項 5 1】 所望のドライバの削除の禁止を設定する削除禁止設定手順を有することを特徴とする請求項 4 9 に記載の情報処理方法。

【請求項 5 2】 前記削除禁止設定手順において、ドライバ削除の禁止が設定されている場合に、前記ドライバ削除手順ではドライバを削除しないように制御する制御手順を有することを特徴とする請求項 5 1 に記載の情報処理方法。

【請求項 5 3】 外部情報処理装置と通信して、セットアップ情報の収集を行う収集手順を有することを特徴とする請求項 4 9 に記載の情報処理方法。

【請求項 5 4】 外部情報処理装置からの要求に応答して、セットアップ情報を返送することを特徴とする請求項 5 3 に記載の情報処理方法。

【請求項 5 5】 前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手順をさらに有し、

前記ドライバ削除手順は、前記判断手順により指定されたドライバがセットアップされていない場合には、前記削除通知を無視することを特徴とする請求項 4 9 に記載の情報処理方法。

【請求項 5 6】 外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力処理と、前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除処理とを情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 5 7】 周辺機器並びに第一の情報処理装置並びに第二の情報処理装置を含むネットワークシステムであって、

前記第一の情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、

前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記第二の情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段と、

ネットワークを介して前記ドライバの削除通知を前記第二の情報処理装置に入力する入力手段と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えたことを特徴とするネットワークシステム。

【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】平成17年5月12日(2005.5.12)

【公開番号】特開2002-251366(P2002-251366A)

【公開日】平成14年9月6日(2002.9.6)

【出願番号】特願2001-355016(P2001-355016)

【国際特許分類第7版】

G06F 13/10

G06F 9/445

【F I】

G06F 13/10 320 A

G06F 9/06 610 B

G06F 9/06 610 D

G06F 9/06 610 C

G06F 9/06 610 L

【手続補正書】

【提出日】平成16年6月29日(2004.6.29)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

周辺機器と通信する情報処理装置であって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、

前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、

前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手段とを備えたことを特徴とする情報処理装置。

【請求項2】

前記判定手段は、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在するか否かを判定することを特徴とする請求項1に記載の情報処理装置。

【請求項3】

前記ドライバ削除手段によるドライバの削除の実行を指示する指示手段を備え、前記ドライバ削除手段は、前記指示手段による一回の指示に応じて、複数のドライバの削除を実行することを特徴とする請求項1に記載の情報処理装置。

【請求項4】

前記ドライバ削除手段により前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項3に記載の情報処理装置。

【請求項5】

所望のドライバの削除の禁止設定の指示を行う削除禁止設定指示手段を備えたことを特徴とする請求項1に記載の情報処理装置。

【請求項6】

前記情報処理装置は外部情報処理装置と通信するものであり、

前記判定手段が、前記ドライバに対応する周辺機器が存在しないと判断し、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドラ

イバ削除通知を行うドライバ削除通知手段をさらに備えたことを特徴とする請求項1に記載の情報処理装置。

【請求項7】

周辺機器と通信する情報処理装置における情報処理方法であって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手順と、

前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手順と

前記判定手順により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手順とを有することを特徴とする情報処理方法。

【請求項8】

前記判定手順では、出力先ポートに問い合わせた応答があるか否かにより周辺機器が存在するか否かを判定することを特徴とする請求項7に記載の情報処理方法。

【請求項9】

前記ドライバ削除手順において、ドライバの削除の実行を指示する指示手順を行い、前記ドライバ削除手順では、前記指示手順による一回の指示に応じて、複数のドライバの削除を実行することを特徴とする請求項7に記載の情報処理方法。

【請求項10】

前記ドライバ削除手順で前記ドライバを削除する前に、削除を実行するか否かをユーザに確認する確認手順を有することを特徴とする請求項7に記載の情報処理方法。

【請求項11】

所望のドライバの削除の禁止の設定を指示を行う削除禁止設定指示手順を有することを特徴とする請求項7に記載の情報処理方法。

【請求項12】

前記情報処理装置は外部情報処理装置と通信するものであり、

前記判定手順で前記ドライバに対応する周辺機器が存在しないと判断され、前記周辺装置がローカルデバイスである場合に、前記外部情報処理装置にドライバを削除させるドライバ削除通知を行うドライバ削除通知手順をさらに有することを特徴とする請求項7に記載の情報処理方法。

【請求項13】

周辺機器と通信する情報処理装置において実行される制御プログラムであって、

前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得処理と、

前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定処理と、

前記判定処理により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除処理とを前記情報処理装置において実行させることを特徴とする制御プログラム。

【請求項14】

周辺機器並びに外部情報処理装置と通信する情報処理装置であって、

セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、

前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段とを備えたことを特徴とする情報処理装置。

【請求項15】

前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段をさらに備え、

前記ドライバ削除通知手段は、前記セットアップ情報に基づいて、ドライバの削除通知を行うことを特徴とする請求項14に記載の情報処理装置。

【請求項 16】

前記ドライバ削除通知手段によるドライバの削除通知の実行を指示する指示手段を備えたことを特徴とする請求項 14 に記載の情報処理装置。

【請求項 17】

前記ドライバ削除通知手段は、前記指示手段による一回の指示に応じて、前記外部情報処理装置に複数のドライバを削除させるドライバ削除通知を行うことを特徴とする請求項 16 に記載の情報処理装置。

【請求項 18】

前記ドライバ削除通知手段により前記ドライバの削除を通知する前に、削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項 14 に記載の情報処理装置。

【請求項 19】

所望のドライバの削除の禁止を設定する指示が可能な削除禁止設定指示手段を備えたことを特徴とする請求項 14 に記載の情報処理装置。

【請求項 20】

前記判定手段が、前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うことを特徴とする請求項 14 に記載の情報処理装置。

【請求項 21】

周辺機器並びに外部情報処理装置と通信する情報処理装置における情報処理方法であって、

セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手順と、

前記判定手順により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手順とを有することを特徴とする情報処理方法。

【請求項 22】

前記外部情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手順をさらに有し、

前記ドライバ削除通知手順は、前記セットアップ情報に基づいて、ドライバの削除通知を行うことを特徴とする請求項 21 に記載の情報処理方法。

【請求項 23】

前記ドライバ削除通知手順によるドライバの削除の通知を指示する指示手順を有することを特徴とする請求項 21 に記載の情報処理方法。

【請求項 24】

前記ドライバ削除通知手順は、前記指示手順による一回の指示に応じて、複数のドライバの削除を前記外部情報処理装置に実行させるドライバ削除通知を行うことを特徴とする請求項 23 に記載の情報処理方法。

【請求項 25】

前記ドライバ削除通知手順により前記ドライバ削除通知を行う前に、前記外部情報処理装置に削除を実行させるか否かをユーザに確認する確認手順を有することを特徴とする請求項 21 に記載の情報処理方法。

【請求項 26】

所望のドライバの削除の禁止を設定する指示が可能な削除禁止設定指示手順を有することを特徴とする請求項 21 に記載の情報処理方法。

【請求項 27】

前記判定手順で前記ドライバに対応する周辺機器が存在しないと判定し、前記周辺機器がローカルデバイスである場合に、前記外部情報処理装置にドライバの削除を行わせるドライバ削除通知を行うことを特徴とする請求項 21 に記載の情報処理方法。

【請求項 28】

周辺機器並びに外部情報処理装置と通信する情報処理装置において実行される制御プログラムであって、

セットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定処理と、

前記判定処理により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記外部情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知処理とを前記情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 29】

周辺装置並びに複数の外部情報処理装置と通信する情報処理装置において、

前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手段と、

前記認識手段により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手段とを備えたことを特徴とする情報処理装置。

【請求項 30】

前記通知手段は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うことを特徴とする請求項 29 に記載の情報処理装置。

【請求項 31】

前記認識手段は、前記ローカルデバイスが存在するか否かを認識する機能、及び、前記ローカルデバイスが共有設定されているか否かを認識する機能を備えたことを特徴とする請求項 29 に記載の情報処理装置。

【請求項 32】

前記通知手段は、前記認識手段により前記ローカルデバイスが存在しないと認識された場合、又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うことを特徴とする請求項 31 に記載の情報処理装置。

【請求項 33】

前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識することを特徴とする請求項 29 に記載の情報処理装置。

【請求項 34】

周辺装置並びに複数の外部情報処理装置と通信する情報処理装置における情報処理方法において、

前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識手順と、

前記認識手順により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知手順とを有することを特徴とする情報処理方法。

【請求項 35】

前記通知手順は、複数の外部情報処理装置に対して一斉にドライバの削除通知を行うことを特徴とする請求項 34 に記載の情報処理方法。

【請求項 36】

前記認識手順は、前記ローカルデバイスが存在するか否かを認識する手順、及び、前記ローカルデバイスが共有設定されているか否かを認識する手順を有することを特徴とする請求項 34 に記載の情報処理方法。

【請求項 37】

前記通知手順は、前記認識手順により前記ローカルデバイスが存在しないと認識された場合、又は、前記ローカルデバイスが共有設定されていないと認識された場合に、前記ドライバの削除通知を行うことを特徴とする請求項 36 に記載の情報処理方法。

【請求項 38】

前記複数の情報処理装置から受信したセットアップ情報に基づいて、前記ローカルデバイスを共有設定している前記複数の情報処理装置を認識することを特徴とする請求項 3 4 に記載の情報処理方法。

【請求項 3 9】

周辺装置並びに複数の外部情報処理装置と通信する情報処理装置における制御プログラムにおいて、

前記情報処理装置に接続されたローカルデバイスの接続状況を認識する認識処理と、

前記認識処理により認識されたローカルデバイスの接続状況により、外部情報処理装置から前記ローカルデバイスを使用できない場合に、外部情報処理装置に対して前記ローカルデバイスに対応するドライバの削除通知を行う通知処理とを前記情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 4 0】

外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手段と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えたことを特徴とする情報処理装置。

【請求項 4 1】

前記ドライバ削除手段により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認手段を備えたことを特徴とする請求項 4 0 に記載の情報処理装置。

【請求項 4 2】

所望のドライバの削除の禁止を設定する指示が可能な削除禁止設定指示手段を備えたことを特徴とする請求項 4 0 に記載の情報処理装置。

【請求項 4 3】

前記削除禁止設定指示手段において、ドライバ削除の禁止が指示されている場合に、前記ドライバ削除手段がドライバを削除しないように制御する制御手段を備えたことを特徴とする請求項 4 2 に記載の情報処理装置。

【請求項 4 4】

外部情報処理装置と通信して、セットアップ情報の収集を行う収集手段を備えたことを特徴とする請求項 4 0 に記載の情報処理装置。

【請求項 4 5】

外部情報処理装置からの要求に応答して、セットアップ情報を返送することを特徴とする請求項 4 3 に記載の情報処理装置。

【請求項 4 6】

前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手段をさらに備え、

前記ドライバ削除手段は、前記判断手段により指定されたドライバがセットアップされていない場合には、前記削除通知を無視することを特徴とする請求項 4 0 に記載の情報処理装置。

【請求項 4 7】

外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力手段と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手順とを有することを特徴とする情報処理方法。

【請求項 4 8】

前記ドライバ削除手順により前記ドライバを削除する前に、前記ドライバの削除を実行するか否かをユーザに確認する確認手順を有することを特徴とする請求項 4 7 に記載の情報処理方法。

【請求項 4 9】

所望のドライバの削除の禁止を設定する指示が可能な削除禁止設定指示手順を有するこ

とを特徴とする請求項 4 7 に記載の情報処理方法。

【請求項 5 0】

前記削除禁止設定指示手順において、ドライバ削除の禁止が設定されている場合に、前記ドライバ削除手順ではドライバを削除しないように制御する制御手順を有することを特徴とする請求項 4 9 に記載の情報処理方法。

【請求項 5 1】

外部情報処理装置と通信して、セットアップ情報の収集を行う収集手順を有することを特徴とする請求項 4 7 に記載の情報処理方法。

【請求項 5 2】

外部情報処理装置からの要求に応答して、セットアップ情報を返送することを特徴とする請求項 5 1 に記載の情報処理方法。

【請求項 5 3】

前記削除通知で指定されたドライバが情報処理装置にセットアップされているか否かを判断する判断手順をさらに有し、

前記ドライバ削除手順は、前記判断手順により指定されたドライバがセットアップされていない場合には、前記削除通知を無視することを特徴とする請求項 4 7 に記載の情報処理方法。

【請求項 5 4】

外部ネットワークを介してドライバの指定を含むドライバの削除通知を入力する入力処理と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除処理とを情報処理装置に実行させることを特徴とする制御プログラム。

【請求項 5 5】

周辺機器並びに第一の情報処理装置並びに第二の情報処理装置を含むネットワークシステムであって、

前記第一の情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、

前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記第二の情報処理装置に前記周辺装置に対応するドライバを削除させるドライバ削除通知を行うドライバ削除通知手段と、

ネットワークを介して前記ドライバの削除通知を前記第二の情報処理装置に入力する入力手段と、

前記削除通知が入力された場合に、指定されたドライバを削除するドライバ削除手段とを備えたことを特徴とするネットワークシステム。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0 0 0 5

【補正方法】変更

【補正の内容】

【0 0 0 5】

【課題を解決するための手段】

上記課題を解決するための手段として、本発明の情報処理装置について述べると、本発明の情報処理装置は、周辺機器と通信する情報処理装置であって、前記情報処理装置におけるドライバのセットアップ状況を取得するセットアップ情報取得手段と、前記情報処理装置にセットアップされているドライバに対応する周辺機器が存在するか否かを判定する判定手段と、前記判定手段により前記ドライバに対応する周辺機器が存在しないと判定された場合に、前記ドライバを削除するドライバ削除手段とを備えた点に特徴を有する。